

iWay

iWay ETL Manager User's Guide
Version 5 Release 2.0

Cactus, EDA/SQL, FIDEL, FOCCALC, FOCUS, FOCUS Fusion, FOCUS Vision, Hospital-Trac, Information Builders, the Information Builders logo, Parlay, PC/FOCUS, SmartMart, SmartMode, SNAPPack, TableTalk, WALDO, Web390, WebFOCUS and WorldMART are registered trademarks and EDA, iWay, and iWay Software are trademarks of Information Builders, Inc.

Acrobat and Adobe are registered trademarks of Adobe Systems Incorporated.

Tuxedo and WebLogic are registered trademarks of BEA Systems, Inc.

Alpha, DEC, DECnet, and NonStop are registered trademarks and Tru64, OpenVMS, and VMS are trademarks of Compaq Computer Corporation.

CA-ACF2, CA-Datcom, CA-IDMS, CA-Top Secret, and Ingres are registered trademarks of Computer Associates International, Inc.

HP MPE/iX is a registered trademark of Hewlett Packard Corporation.

Informix is a registered trademark of Informix Software, Inc.

Intel and Pentium are registered trademarks of Intel Corporation.

ACF/VTAM, AIX, AS/400, CICS, DB2, DRDA, Distributed Relational Database Architecture, IBM, MQSeries, MVS/ESA, OS/2, OS/390, OS/400, RACF, RS/6000, S/390, VM/ESA, VSE/ESA, VTAM, and WebSphere are registered trademarks and DB2/2, HiperSpace, IMS, iSeries, MVS, QMF, SQL/DS, z/OS and z/VM are trademarks of International Business Machines Corporation.

Approach and DataLens are registered trademarks of Lotus Development Corporation.

ObjectView is a trademark of Matesys Corporation.

ActiveX, BizTalk, FrontPage, Microsoft, MS-DOS, PowerPoint, Visual Basic, Visual C++, Visio, Visual FoxPro, Windows, and Windows NT are registered trademarks of Microsoft Corporation.

Teradata is a registered trademark of NCR International, Inc.

Netscape, Netscape FastTrack Server, and Netscape Navigator are registered trademarks of Netscape Communications Corporation.

ServletExec is a trademark of New Atlanta Communications, LLC.

NetWare and Novell are registered trademarks of Novell, Inc.

Oracle is a registered trademark and Rdb is a trademark of Oracle Corporation.

PeopleSoft is a registered trademark of PeopleSoft, Inc.

INFOAccess is a trademark of Pioneer Systems, Inc.

Progress is a registered trademark of Progress Software Corporation.

Red Brick Warehouse is a trademark of Red Brick Systems.

Red Hat, RPM, Maximum RPM, Linux Library, PowerTools, Linux Undercover, RHmember, RHmember More, Rough Cuts, and Rawhide are trademarks or registered trademarks of Red Hat, Inc. in the United States and other countries.

SAP and SAP R/3 are registered trademarks and SAP Business Information Warehouse and SAP BW are trademarks of SAP AG.

Siebel is a trademark of Siebel Systems, Inc.

Reliant UNIX is a registered trademark of Fujitsu Siemens Computers.

Silverstream is a trademark of Silverstream Software.

ADABAS is a registered trademark of Software A.G.

CONNECT:Direct is a trademark of Sterling Commerce.

Java and all Java-based marks, NetDynamics, Solaris, SunOS, and iPlanet are trademarks or registered trademarks of Sun Microsystems, Inc. in the U.S. and other countries.

PowerBuilder, Powersoft, and Sybase are registered trademarks and SQL Server is a trademark of Sybase, Inc.

Unicode is a trademark of Unicode, Inc.

UNIX is a registered trademark of The Open Group in the United States and other countries.

SWIFT is trademark of Synopsys, Inc.

Due to the nature of this material, this document refers to numerous hardware and software products by their trade names. In most, if not all cases, these designations are claimed as trademarks or registered trademarks by their respective companies. It is not this publisher's intent to use any of these names generically. The reader is therefore cautioned to investigate all claimed trademark rights before using any of these names other than to refer to the product described.

Copyright © 2003, by Information Builders, Inc. All rights reserved. This manual, or parts thereof, may not be reproduced in any form without the written permission of Information Builders, Inc.

Printed in the U.S.A.

Preface

This documentation describes how to use/install ETL Manager. It is intended for the person responsible for transforming data by designing and executing ETL Requests.

How This Manual Is Organized

This manual includes the following chapters:

Chapter/Appendix		Contents
1	<i>Introduction to ETL Manager</i>	Describes the purpose of data warehousing, and the general concepts and processes of ETL Manager.
2	<i>Installing ETL Manager</i>	Describes how to install ETL Manager.
3	<i>Getting Started</i>	Describes how to begin using ETL Workbench.
4	<i>Building an ETL Request</i>	Describes how to create and build a new ETL Request.
5	<i>Scheduling and Executing ETL Requests</i>	Describes how to schedule an ETL Request with the Scheduler tool and Dependencies tool.
6	<i>ETL Manager Tutorial</i>	Helps you create a sample ETL Request.
7	<i>Working With an ETL Request</i>	Provides instructions on managing an ETL Request.
8	<i>Controlling ETL Requests With Procedures</i>	Describes how to use procedures to perform tasks that are not automated by ETL Manager.
9	<i>Generating Reports and Viewing the Log File</i>	Enables you to view reports on how your ETL Requests run, what the ETL Server environment looks like, and what error messages the ETL Server generates.
10	<i>Data Management</i>	Describes the data sources, data targets, and Synonyms used in ETL Manager.
11	<i>ETL Manager Security</i>	Describes issues related to establishing security for ETL Manager.
12	<i>Using the iWay Web Console</i>	Describes how to configure ETL Manager with the iWay Web Console.

Chapter/Appendix		Contents
13	<i>Improving Performance</i>	Provides tips on how to insure successful deployment of your data mart and data warehouse implementations.
A	<i>ETL Manager Internal Data Store</i>	Provides information about the ETL Manager internal data store.
B	<i>Problems, Errors, and Troubleshooting</i>	Assists you in resolving problems that may occur during the development of your ETL Manager application.

Documentation Conventions

The following conventions apply throughout this manual:

Convention	Description
THIS TYPEFACE or <i>this typeface</i>	Denotes syntax that you must enter exactly as shown.
<i>this typeface</i>	Represents a placeholder (or variable) in syntax for a value that you or the system must supply.
<u>underscore</u>	Indicates a default setting.
<i>this typeface</i>	Represents a placeholder (or variable) in a text paragraph, a cross-reference, or an important term.
this typeface	Highlights a file name or command in a text paragraph that must be lowercase.
<i>this typeface</i>	Indicates a button, menu item, or dialog box option you can click or select.
Key + Key	Indicates keys that you must press simultaneously.
{ }	Indicates two or three choices; type one of them, not the braces.
[]	Indicates a group of optional parameters. None are required, but you may select one of them. Type only the parameter in the brackets, not the brackets.
	Separates mutually exclusive choices in syntax. Type one of them, not the symbol.

Convention	Description
...	Indicates that you can enter a parameter multiple times. Type only the parameter, not the ellipsis points (...).
.	Indicates that there are (or could be) intervening or additional commands.

Related Publications

Visit our World Wide Web site, <http://www.iwaysoftware.com>, to view a current listing of our publications and to place an order. You can also contact the Publications Order Department at (800) 969-4636.

Customer Support

Do you have questions about ETL Manager?

Call Information Builders Customer Support Service (CSS) at (800) 736-6130 or (212) 736-6130. Customer Support Consultants are available Monday through Friday between 8:00 a.m. and 8:00 p.m. EST to address all your ETL Manager questions. Information Builders consultants can also give you general guidance regarding product capabilities and documentation. Please be ready to provide your six-digit site code (xxxx.xx) when you call.

You can also access support services electronically, 24 hours a day, with InfoResponse Online. InfoResponse Online is accessible through our World Wide Web site, <http://www.informationbuilders.com>. It connects you to the tracking system and known-problem database at the Information Builders support center. Registered users can open, update, and view the status of cases in the tracking system and read descriptions of reported software issues. New users can register immediately for this service. The technical support section of www.informationbuilders.com also provides usage techniques, diagnostic tips, and answers to frequently asked questions.

To learn about the full range of available support services, ask your Information Builders representative about InfoResponse Online, or call (800) 969-INFO.

Information You Should Have

To help our consultants answer your questions most effectively, be ready to provide the following information when you call:

- Your six-digit site code (xxxx.xx).
- Your ETL Manager configuration:

- The ETL Workbench release and build number.
- The iWay Server release.
- The ETL Request.
- The Master Files and Access Files for all data sources and data targets used.
- If a data source is used, the data source name and release level.
- If a procedure is used, the .FEX file for the procedure.
- For problems occurring when opening, building, or saving an ETL Request:
 - The exact nature of the problem.
 - Any error messages or error codes.
 - The steps that can be used to reproduce the problem.
- For problems occurring while running a request, the ETL Request log.
- For Server problems, a server trace. Acquire this by starting the server with traces and reproducing the problem.

User Feedback

In an effort to produce effective documentation, the Documentation Services staff welcomes any opinion you can offer regarding this manual. Please use the Reader Comments form at the end of this manual to relay suggestions for improving the publication or to alert us to corrections. You can also use the Documentation Feedback form on our Web site, <http://www.iwaysoftware.com>.

Thank you, in advance, for your comments.

iWay Software Training and Professional Services

Interested in training? Our Education Department offers a wide variety of training courses for iWay Software and other Information Builders products.

For information on course descriptions, locations, and dates, or to register for classes, visit our World Wide Web site (<http://www.iwaysoftware.com>) or call (800) 969-INFO to speak to an Education Representative.

Interested in technical assistance for your implementation? Our Professional Services department provides expert design, systems architecture, implementation, and project management services for all your business integration projects. For information, visit our World Wide Web site (<http://www.iwaysoftware.com>).

Contents

1. Introduction to ETL Manager	1-1
ETL Management and Data Migration Overview	1-2
What is ETL Manager?	1-3
Using ETL Requests	1-3
ETL Manager Components	1-3
ETL Workbench	1-4
The Server	1-5
2. Installing ETL Manager	2-1
Installation and Configuration Overview	2-2
Configuring the Server	2-2
Installing ETL Manager Workbench	2-3
Upgrading ETL Manager	2-11
3. Getting Started	3-1
The ETL Workbench Main Window	3-2
The ETL Workbench Main Toolbar	3-3
Connecting to the Server	3-4
Viewing Server Properties	3-5
Using the Message Window	3-6
Ending an ETL Workbench Session	3-8
Getting Help	3-8
Information Builders on the Web	3-8
4. Building an ETL Request	4-1
Beginning an ETL Request	4-2
Creating a New ETL Request	4-2
Setting Properties of an ETL Request	4-2
Building an ETL Request	4-5
Using the Workflow Tool	4-6
Adding and Deleting a Component	4-7
Connecting and Disconnecting Components in Your ETL Request	4-7
Opening and Closing a Component	4-8
Selecting Fields in a Component	4-8
Identifying Data Sources	4-9
Updating the List of Data Sources	4-13
Creating an Extract Transform	4-13
Specifying Joins	4-17
Specifying a Filter	4-22
Creating a Sub-Query in a Filter Condition	4-28
Using a Variable in a Filter	4-28

Selecting Columns	4-30
Aggregating Columns	4-34
Creating an SQL Calculation	4-34
Specifying a Sort	4-39
Testing the SQL	4-42
Specifying Transport Options	4-44
Selecting a Data Target	4-46
Updating the List of Data Targets	4-53
Specifying Mappings	4-56
Creating a Temporary Column for a Mapping	4-60
Using the Transform Calculator	4-60
Field Formats and Mapping	4-72
Converting Legacy Character Date Fields Into Date Format	4-76
Validating Records	4-77
Specifying Data Target Options	4-77
Specifying Target Setup Options	4-77
Setting Up Output Parameters	4-83
Using Bulk Loading	4-89
Specifying Alternate Files for Bulk Loading	4-92
Selecting Run Options	4-93
Setting Additional Options for Oracle, DB2, Nucleus, and Teradata	4-95
Using Bulk Load Command and Control Files	4-101
Logging Records	4-102
5. Scheduling and Executing ETL Requests	5-1
Scheduling an ETL Request	5-2
Restarting a Failed ETL Request	5-5
Creating Dependencies	5-5
Event-Based Scheduling	5-7
Starting ETL Requests With CMRUN	5-8
Starting ETL Requests With CMASAP	5-13
Starting ETL Requests With Your Own API Program	5-15
Starting ETL Requests From FOCUS	5-17
Determining the Status of an Event-Based ETL Request	5-18

6. ETL Manager Tutorial	6-1
Before You Begin	6-2
Building the ETL Request	6-2
Selecting Data Sources	6-2
Joining Data Sources	6-8
Adding Filters to the ETL Request	6-10
Selecting Columns	6-12
Specifying a Sort	6-15
Testing the SQL	6-17
Identifying the Target Table and Specifying Mappings	6-19
Specifying Load Options	6-24
Scheduling and Running the ETL Request	6-27
Scheduling the ETL Request	6-27
Running the ETL Request	6-28
Viewing the ETL Request Log	6-28
7. Working With an ETL Request	7-1
Editing an ETL Request	7-2
Saving an ETL Request	7-3
Printing an ETL Request	7-4
Copying an ETL Request	7-4
Deleting an ETL Request	7-5
Failing to Delete an ETL Request	7-6
Exporting an ETL Request	7-6
Executing an ETL Request Immediately	7-7
Specifying ETL Request Properties	7-8
Filtering the Display of ETL Requests	7-9
8. Controlling ETL Requests With Procedures	8-1
Executing a Procedure With an ETL Request	8-2
Executing a Procedure Without an ETL Request	8-5
Running a Stored Procedure	8-5
Viewing Files With EDAGET	8-7
Changing Run-Time Behavior With CMOPTION	8-7
9. Generating Reports and Viewing the Log File	9-1
Running a Report	9-2
Working With a Report	9-4
Using the Report Browser Toolbar	9-5
Saving a Report	9-5
Copying a Report	9-6
Printing a Report	9-6
Changing Report Font	9-6
Viewing the Log File	9-7

10. Data Management	10-1
Storing ETL Requests	10-2
Application Directories	10-2
Using a Data Source	10-2
Naming a Data Source	10-4
Using a Data Target	10-4
Local Relational Data Targets	10-5
Local Non-Relational Files	10-6
Server Transfer Files	10-6
Remote Destination	10-7
Naming a Data Target	10-7
Using a Synonym	10-8
11. ETL Manager Security	11-1
ETL Request Execution and User IDs	11-2
Request Submission	11-2
Data Targets and User Access	11-2
Restricting the Application Paths Available to a User	11-2
12. Using the iWay Web Console	12-1
Starting ETL Manager	12-2
Configuring ETL Manager Behavior	12-3
Configuring the Application Path	12-7
Controlling the ETL Scheduler	12-10
Cancelling ETL Requests	12-10
Refreshing Information in the Web Console	12-10
Working With ETL Requests in the Web Console	12-10
Viewing an ETL Request	12-11
Running an ETL Request	12-11
Running an ETL Request as Deferred	12-11
Deleting ETL Requests	12-12
Viewing ETL Request Run-Time Information	12-12
Viewing an ETL Request's Log	12-13
Viewing the Statistics and Message Logs	12-14
Migrating ETL Requests	12-14
13. Improving Performance	13-1
Designing Your Application	13-2
Reviewing Your Data Sources	13-3
Reviewing Data Targets	13-4
Customizing Your Configuration	13-4
Connection, Tools, and Security	13-5
Naming Target Tables and Target Types	13-5
Accessing Target Tables	13-6

Prototyping	13-7
Improving ETL Manager's Performance	13-8
Improving Data Extraction Performance	13-8
Using Automatic Passthrough	13-9
Improving Data Transit Performance	13-12
Improving Data Loading Performance	13-12
Representing a Null Value in a Flat File and With Bulk Load	13-13
Customizing Your Environment	13-13
Creating a DEFINE in the Master File	13-15
A. ETL Manager Internal Data Store	A-1
Log and Statistics Tables	A-2
ETLLOG Table	A-2
ETLSTATS Table	A-2
Maintaining the Server Log and Statistics Table	A-4
B. Problems, Errors, and Troubleshooting	B-1
ETL Manager Workbench Status Codes and Server Messages	B-2
Server Messages	B-2
iWay Connector Communications Errors	B-3
Server Run-Time Error Codes	B-4
ETL Management Problem Resolution	B-5
Server Message Codes	B-6
Including Run-Time Messages in the Server Log	B-9
Performing a Trace to Diagnose Server Problems	B-9

CHAPTER 1

Introduction to ETL Manager

Topics:

- ETL Management and Data Migration Overview
- What is ETL Manager?
- ETL Manager Components

This topic describes the purpose of data warehousing, and the general concepts and processes of ETL Manager.

ETL Management and Data Migration Overview

ETL management and data migration make up the process of extracting data from one or more *data sources* and copying that data into a data target.

- A **data source** often contains operational data, such as sales, orders, customers, invoices, and inventories. The data source is designed for online transaction processing and high-speed updates. Systems that maintain these data sources are often highly volatile and mission critical. There may be many such systems in a company containing related information with no means for cross-structure comparison.
- A **data target**, referred to as the data warehouse or data mart, is often relational and is generally located on a machine with optimal performance and availability. This data target typically stores information historically and is geared for high performance online analysis.

Often, the process of ETL management involves data “cleansing,” such as numeric aggregation, improvements to cryptic names, or the calculation of new columns. There are many benefits to data warehousing for this process. They include:

- **Ease of use.** The data warehouse may be optimized for analysis, and the data cleansing process ensures that the data is stored in a more readable format.
- **Reduced load on legacy systems.** Since your analysts are no longer accessing your operational data sources, production systems are not impacted by complex queries.
- **Better response time.** The data warehouse can be situated on a local department machine. You no longer have to crisscross through the network to find your data.
- **Ownership.** Many departmental data warehouses can be created, each one containing summary information for that department only.

What is ETL Manager?

ETL Manager automates the process of building a data warehouse with the use of ETL Requests. ETL Manager enables you to do the following:

- Access source data in numerous formats from 14 operating systems.
- Integrate multiple data sources into a single data warehouse.
- Apply powerful data cleansing rules and transformation logic.
- Aggregate data and create roll-ups to aid decision support.
- Use specialized high-volume data movers and loaders.
- Schedule data source updates at user-defined intervals.
- Support scheduled synchronization of data sources.
- View comprehensive logging and transaction statistics.

Since ETL Manager is a true client/server middleware solution, underlying communications protocols and data source subsystems are transparent.

Using ETL Requests

ETL Requests perform the copy management and data warehousing tasks in ETL Manager. The following steps are required to use an ETL Request:

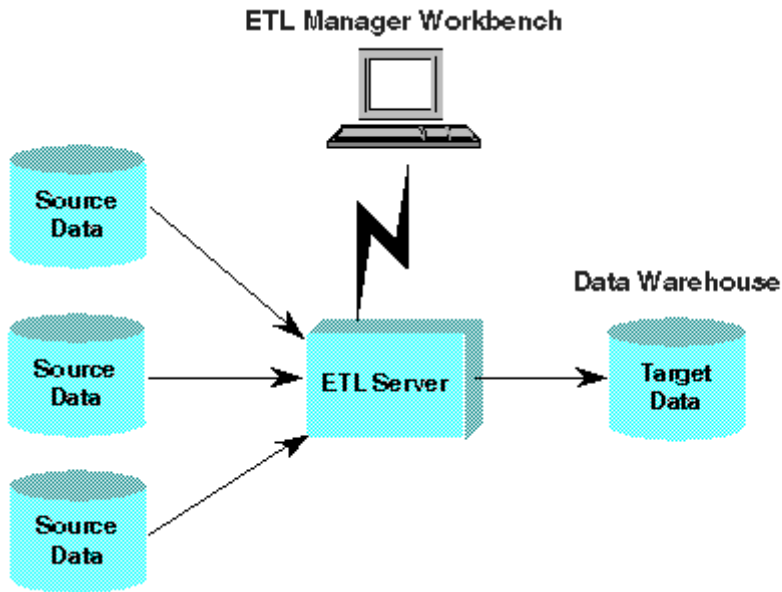
1. Create an ETL Request. For details, see Chapter 4, *Building an ETL Request*.
2. Schedule and execute the ETL Request. For details, see Chapter 5, *Scheduling and Executing ETL Requests*.
3. Optionally, customize your ETL Request. For details, see Chapter 8, *Controlling ETL Requests With Procedures*.

ETL Manager Components

ETL Manager is comprised of two components:

- The **ETL Workbench** is a graphical user interface that runs in Windows NT 4.0 (Service Pack 6 or later), Windows 2000 and Windows XP. You can use the Workbench to design your query, test it, and establish run-time parameters.
- An **iWay Hub Server** with the Server installed manages execution of the query.

The following figure illustrates an overview of ETL Manager:



ETL Workbench

The ETL Workbench is a Microsoft Windows application that is responsible for building queries, gathering data target information, constructing data transformations, and arranging schedules. It contains the following tools:

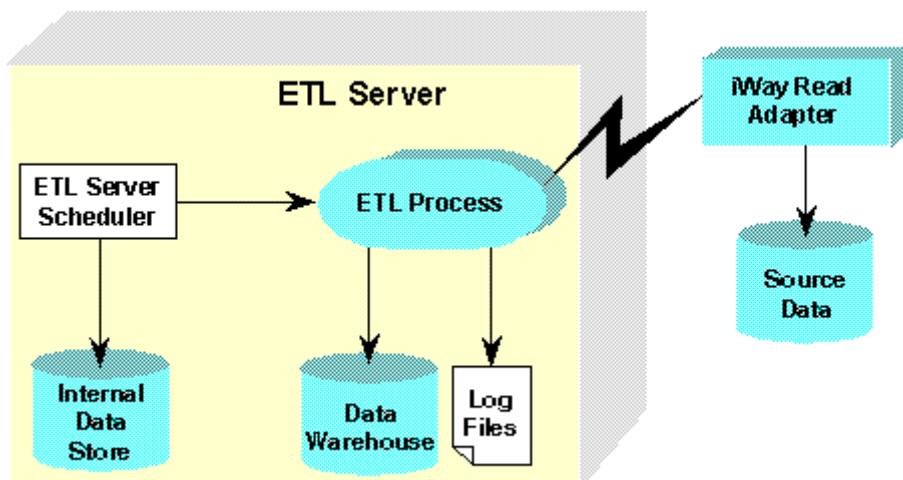
- **The Workflow tool** is a workspace in which you can create your ETL Request by adding data sources, joins, filters, sorts, mappings, and a data target.
- **The Scheduler tool** enables you to select the date, time, and repetitive interval for ETL Request execution.
- **The Dependencies tool** enables you to schedule an ETL Request to execute immediately when another ETL Request finishes executing.
- **The Procedures tool** allows you to specify remote procedures to run as part of the ETL Request.

The Server

The Server is made up of the following components:

- **The internal data store** is maintained by the ETL Workbench. It contains all ETL Manager configuration and request related information, as well as individual ETL Request run-time statistics. The Server accesses the internal data store to schedule, run, and control ETL Requests.
- **The Scheduler** “wakes up” every five or fifteen minutes and reads the internal data store to locate ETL Requests scheduled to run during that interval. For those ETL Requests, the Scheduler concurrently executes the Extractor/Loader. After all scheduled ETL Requests have been executed, the Scheduler updates the internal data store with appropriate statistics and goes “back to sleep” until the next time interval.
- **The Extractor/Loader** performs the following tasks:
 - Reads the internal data store for ETL Request details such as security rules, extraction information, query information, and target warehouse parameters.
 - Prepares the data target to receive data. If the table exists, data will be populated dependent on the append/update rules specified in the ETL Workbench. If the requested table does not exist, the Extractor/Loader correctly formats it in the syntax of the relational data source system.
 - Executes the query and retrieves rows a block at a time. Each retrieved row will be written into the target table in a manner dependent on the insert logic specified in the ETL Workbench (there is different logic for append, insert with update, insert with reject, insert with replace, and so on). Where available, the Extractor/Loader uses Bulk RDBMS APIs for the loading of target new tables (for the supported RDBMS).
 - Populates log files with errors and significant event information such as ETL Request progress and completion statistics, read, write, security, and RDBMS errors.

The following figure illustrates the ETL Manager Server environment:



CHAPTER 2

Installing ETL Manager

Topics:

- Installation and Configuration Overview
- Configuring the Server
- Installing ETL Manager Workbench
- Upgrading ETL Manager

This topic describes how to install ETL Manager and upgrade from a previous version.

Installation and Configuration Overview

Follow these steps to install and configure ETL Manager and its components:

1. Install the Server. For details, see *iWay Server for UNIX, Windows, OpenVMS, OS/400, UNIX System Services Configuration and Operation*.
2. Configure the Server. For details, see *Configuring the Server* on page 2-2.
3. Install the iWay Connector. You must install the iWay Connector before you can install ETL Manager. For details, see the *iWay Connector Suite Installation for Windows NT/2000*.
4. Install ETL Manager. For details, see *Installing ETL Manager Workbench* on page 2-3.

If you are upgrading ETL Manager from a previous release, see *Upgrading ETL Manager* on page 2-11.

Configuring the Server

After installing the Server, you must configure the Server from the Web Console. Until you configure the ETL Manager internal tables, the Scheduler will not start.

Procedure How to Configure the Server

1. Go to the directory in which you installed the server:
 - In Windows NT/2000, select the *iWay 52 ETL Manager group* from the Start menu, and select *Workspace Web Console*.
 - In other systems, or for a UNIX server, start a Web browser and connect to the server using the next consecutive port number after the one you specified during installation.
2. Click *Login as*, and enter an administrator user name and password in the window that appears.
3. Click *login*.
4. On the Web Console main window, click the *ETL* link.
The ETL Configuration window opens.
5. In the ETL Log and Statistics section, click *Create/Recreate*.
6. Click *Submit*.

Installing ETL Manager Workbench

The files required for installing ETL Manager are available on the iWay Data Management Administration Tools CD-ROM.

If you have previously installed Copy Manager or other tools from the EDA Administration Tools CD, you must first uninstall it before installing ETL Manager 5.1. For details, see *How to Remove a Previous Version of Administration Tools* on page 2-3.

Reference Requirements for Installing ETL Workbench

To install ETL Workbench, you need the following:

- An IBM or IBM-compatible computer with a Pentium 233MHz chip or higher.
- A mouse or compatible pointing device supported by Windows.
- At least 264 MB of total memory.
- A hard drive with at least 15 MB of free disk space.
- One of the following operating systems:
 - Windows NT 4.0 (Service Pack 6 or later)
 - Windows 2000
- A CD-ROM drive (for installation only).
- SVGA or higher resolution graphics card set to at least 800 X 600 pixels and high color (16 bit).
- A network (TCP/IP, LU6.2) connecting to the Server.

Procedure How to Remove a Previous Version of Administration Tools

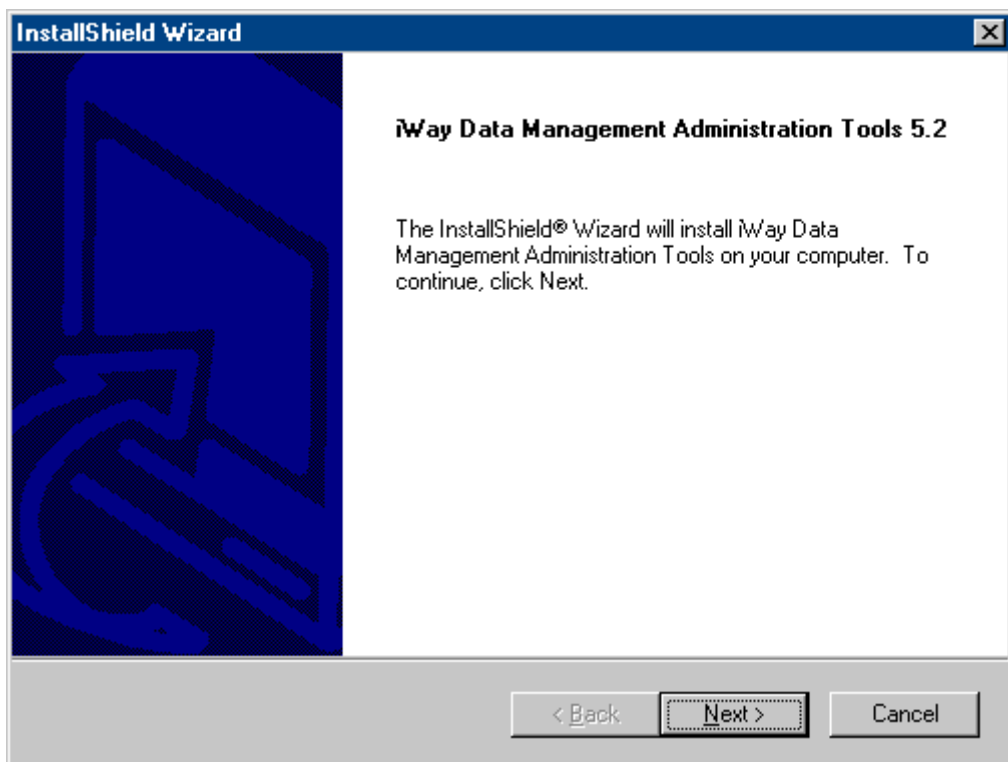
1. Select *Add/Remove Programs* from the Control Panel.
2. Select *iWay Administration* tools from the list, and click *Remove*.
The InstallShield Wizard opens.
3. Select *Remove*, and click *Next>*.
The Confirm File Deletion window opens.
4. Click *OK*.

Administration Tools is removed.

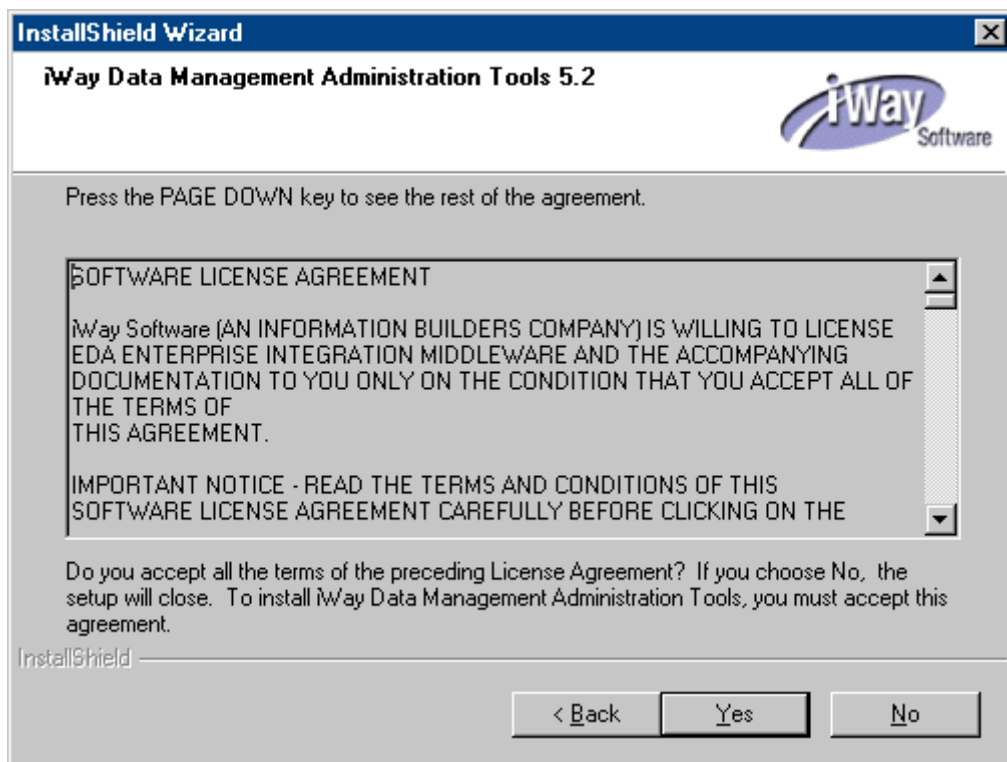
Procedure **How to Install ETL Manager**

1. Close any open programs.
2. Insert the Data Management Administration Tools CD-ROM in your CD-ROM drive. The setup program will start automatically. If it does not, run SETUP.EXE.

The setup program begins and the InstallShield Wizard opens.



3. Click *Next>* to begin the install procedure. The license agreement appears.

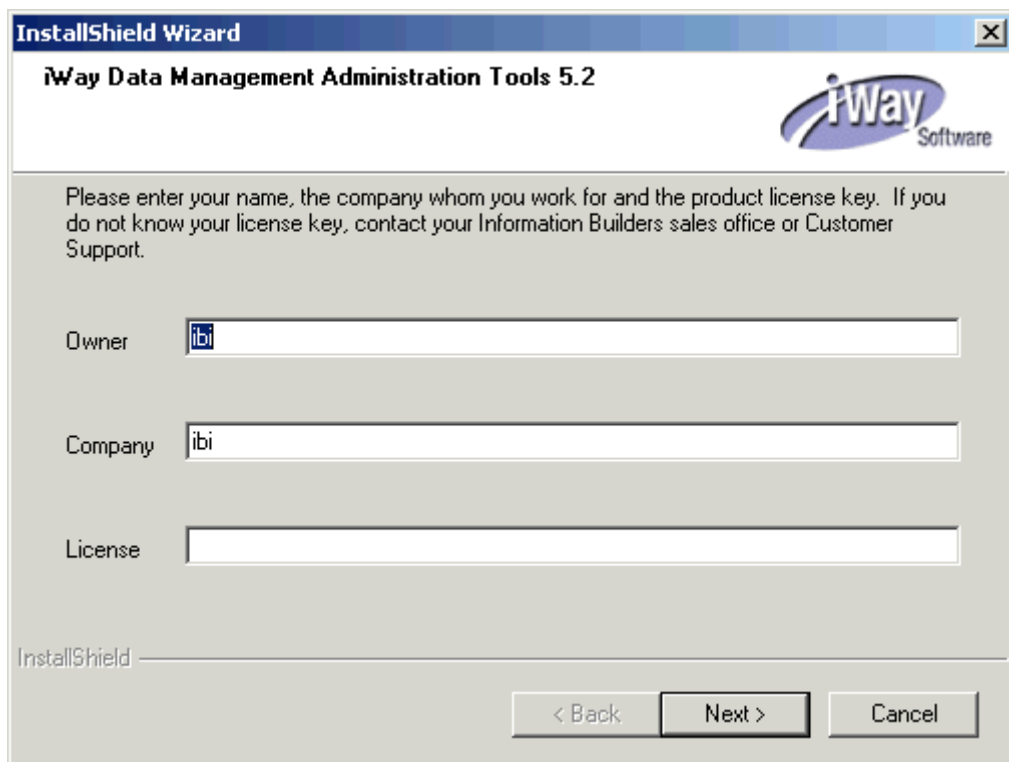


Note: The Data Management Administration Tools CD checks to see if you already have a version of the Administration Tools installed. If you do, you are instructed to remove the old version before installing a new one. You cannot install ETL Manager with a previous version already installed. For details on removing ETL Manager, see *How to Remove a Previous Version of Administration Tools* on page 2-3.

4. Click *Yes* to accept the terms of the license agreement.

If you do not accept the terms of this agreement, click *No* to exit the InstallShield Wizard.

After clicking Yes, the software registration window opens:

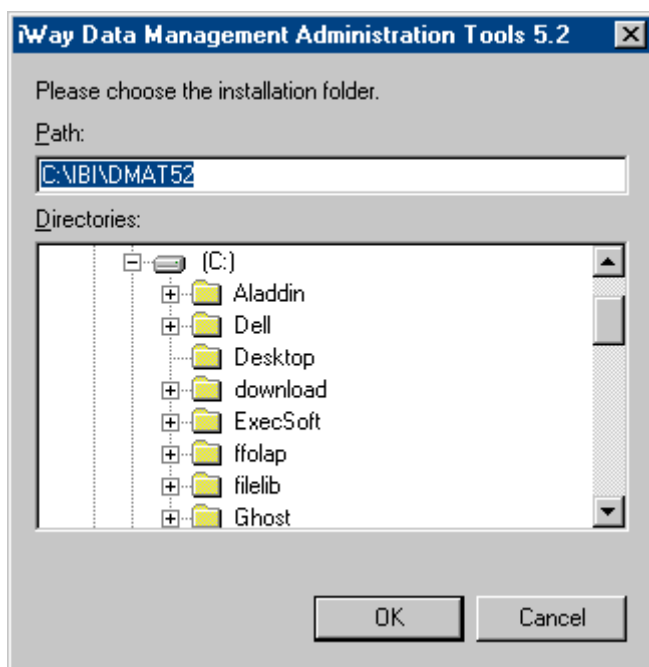


The image shows a software registration window titled "InstallShield Wizard" for "iWay Data Management Administration Tools 5.2". The iWay Software logo is in the top right corner. The main text asks the user to enter their name, company, and license key, with a note to contact sales or support if the license key is unknown. There are three input fields: "Owner" (containing "ibi"), "Company" (containing "ibi"), and "License" (empty). At the bottom, there are three buttons: "< Back", "Next >", and "Cancel". The "InstallShield" logo is visible in the bottom left corner of the window.

5. Enter your user name, company name, and license number in the corresponding fields, and click *Next>*.

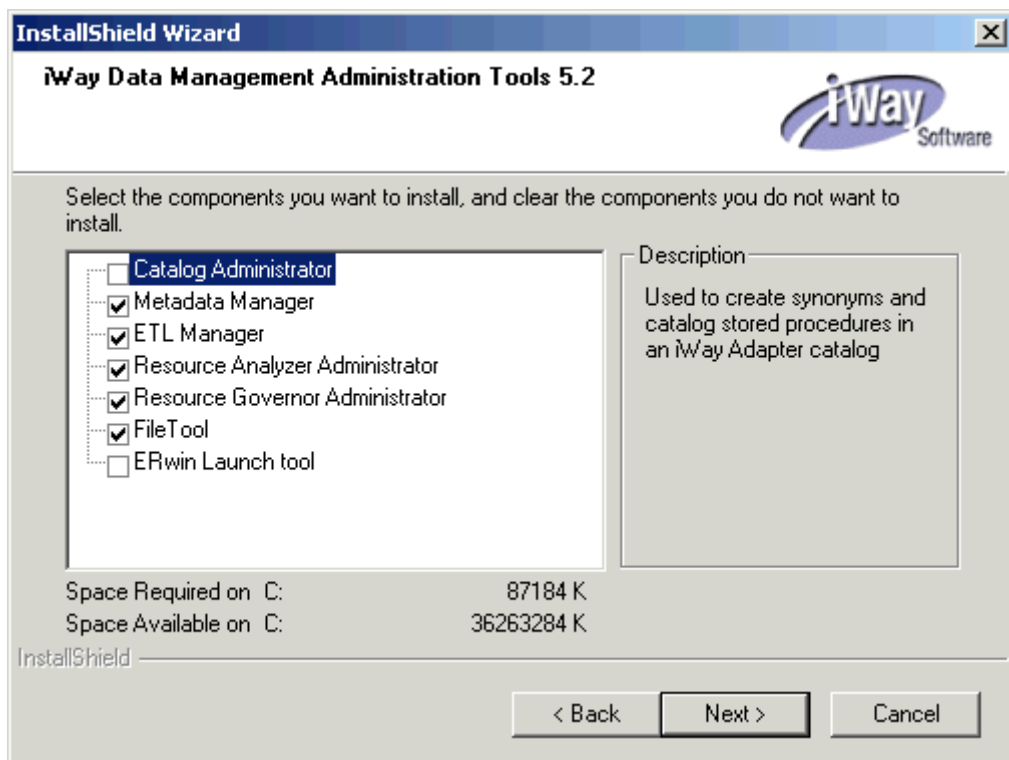
Your license number is found on the packing list you received with your Data Management Administration Tools CD.

The next InstallShield Wizard window opens listing the destination folder:



6. Click *Next>* to select the default file location, or use the directory tree to choose another location.

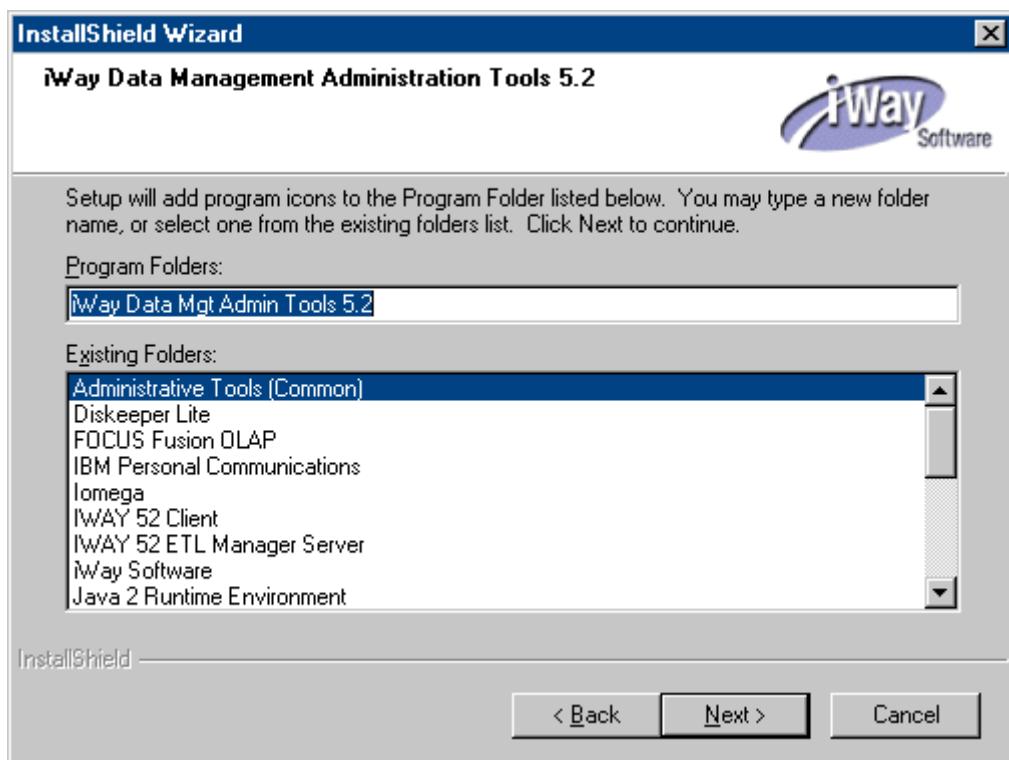
The next InstallShield Wizard window opens, and lists the components that will be installed selected.



7. Ensure the components you want to install are selected and click *Next>* to confirm the components that will be installed.

You need to install ETL Manager, and it is recommended you also install Catalog Administrator to manage the creation of Synonyms on your server, and Metadata Manager to manage ETL and iWay metadata.

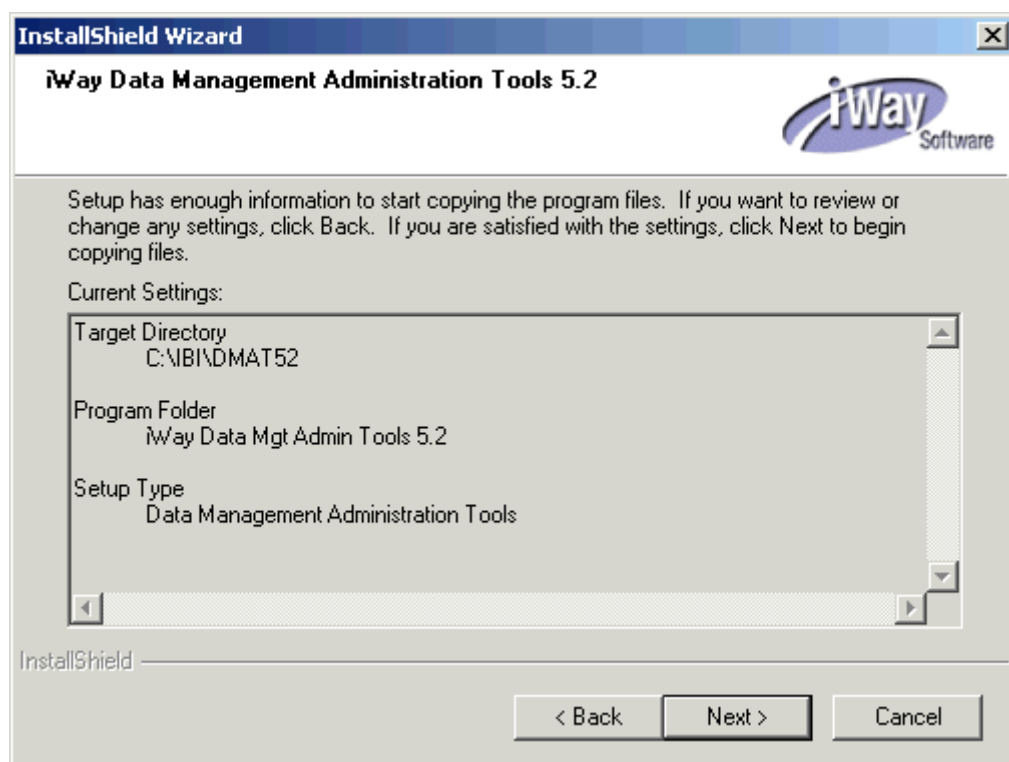
The next InstallShield Wizard window displays the program folder.



8. Click **Next>** to select the default program folder.

If you do not want to accept the default program folder, type a new folder name or select a folder from the existing folders list.

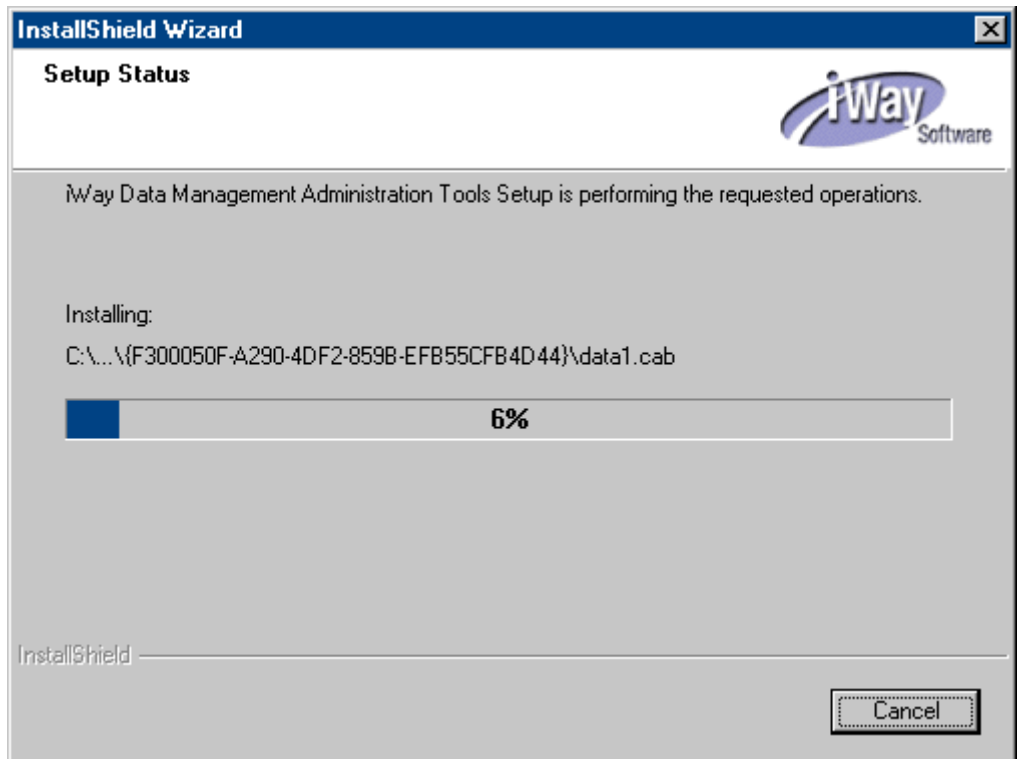
The next InstallShield Wizard window opens, confirming your choices for the target directory, program folder, and setup type.



9. Click *Next>* to accept your choices.

If you want to change your choices, click *<Back* until you reach the appropriate screen, and make your changes.

The installation begins, and the Setup Status window opens.



- 10.** When installation is complete, the InstallShield Wizard Complete window opens, and asks if you want to restart your computer. Select an option, and click *Finish*.

Note: You must restart your computer before using ETL Manager.

Upgrading ETL Manager

When you upgrade ETL Manager from a previous release, you must migrate the Master and Access Files and ETL Requests you want to keep. Master Files and Access Files must be migrated before migrating ETL Requests.

For details, see *Migrating ETL Requests* in Chapter 12, *Using the iWay Web Console*.

CHAPTER 3

Getting Started

Topics:

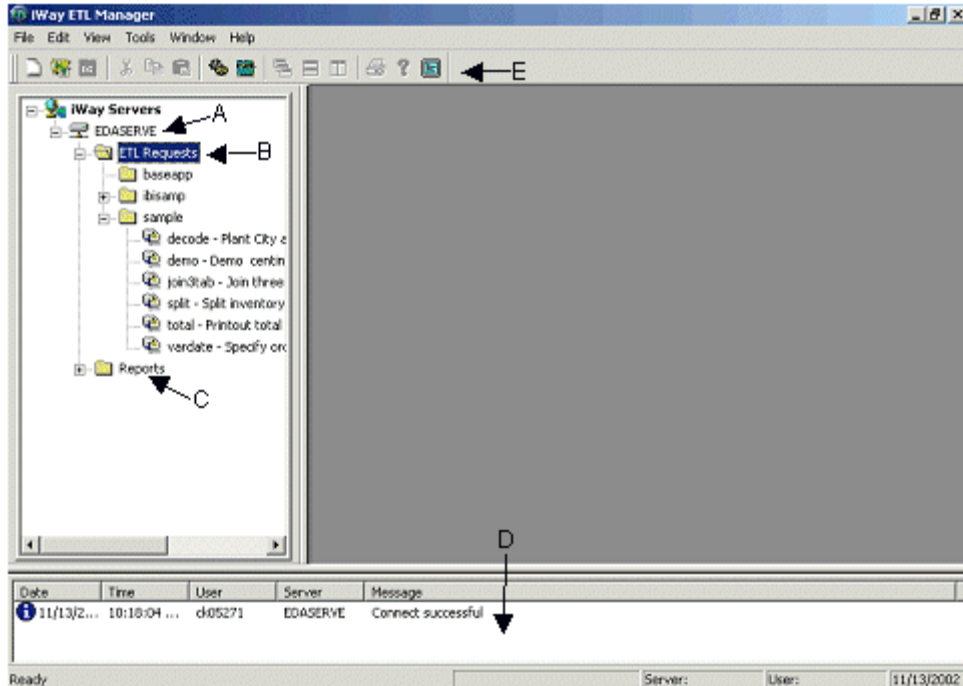
- The ETL Workbench Main Window
- Connecting to the Server
- Viewing Server Properties
- Using the Message Window
- Ending an ETL Workbench Session
- Getting Help
- Information Builders on the Web

This topic describes how to start using ETL Workbench.

The ETL Workbench Main Window

When you open ETL Workbench, the main window opens. From here you can log onto a server, open and create ETL Requests, and open Reports.

The following is the ETL Manager main window:



The ETL Manager main window contains the following:

- A. A list of servers** that can be accessed by ETL Manager and on which you can store your ETL Requests. You designate these Servers in your *odin.cfg* file.













When you are logged on to an iWay Adapter, you can view the ETL Requests and Reports folder in this window. Until you log onto a server, only the servers will display. For details on logging onto a server, see *Connecting to the Server* on page 3-4.


- B. The ETL Requests folder** contains a list of folders representing the directories in which ETL Requests are stored. The ETL Requests can be viewed by opening the directory's folder in which it is stored.
- C. The Reports folder** contains the reports you can run.

- D. **The message window** displays application and server messages. Each message displays the date, time, user, and Server.
- E. **The toolbar** represents the most common activities in ETL Workbench.

The ETL Workbench Main Toolbar

The buttons on the ETL Workbench main toolbar represent the most common activities in ETL Workbench. The toolbar contains the following buttons:

Button	Function
	Creates a new ETL Request.
	Opens an existing ETL Request.
	Saves any changes that you have made to an existing ETL Request. Note: This button is grayed out when there are no changes to be saved.
	Cuts an existing ETL Request file and moves it to the clipboard. It is deleted when the ETL Request is pasted.
	Copies an existing ETL Request file.
	Pastes an existing ETL Request into an open application.
	Allows you to change report fonts and filter messages.
	Runs a remote procedure.
	Tiles open windows.
	Cascades open windows horizontally.
	Cascades open windows vertically.
	Prints the open file.

Button	Function
	Opens ETL Manager's online help system.

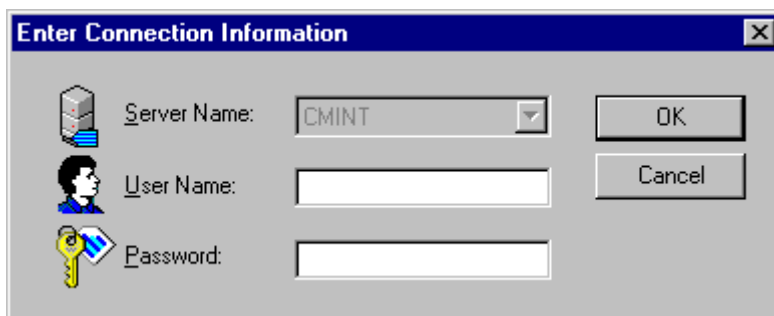
Connecting to the Server

You can connect to the Server from the ETL Workbench main window.

Procedure How to Connect to the Server

1. Do one of the following:
 - Double-click the Server you want to log on to.
or
 - Right-click the server you want to connect to and select *Connect* from the pop-up window.

The Enter Connection Information window opens:



The dialog box titled "Enter Connection Information" has a blue title bar with a close button. It contains three input fields with corresponding icons: a server icon for "Server Name" (containing "CMINT"), a user icon for "User Name", and a key icon for "Password". There are "OK" and "Cancel" buttons on the right.

2. If your Server has security enabled, enter your user name and password in the User Name and Password fields. Windows NT/200 requires your user name be no longer than 32 characters.
3. Click *OK*.

Viewing Server Properties

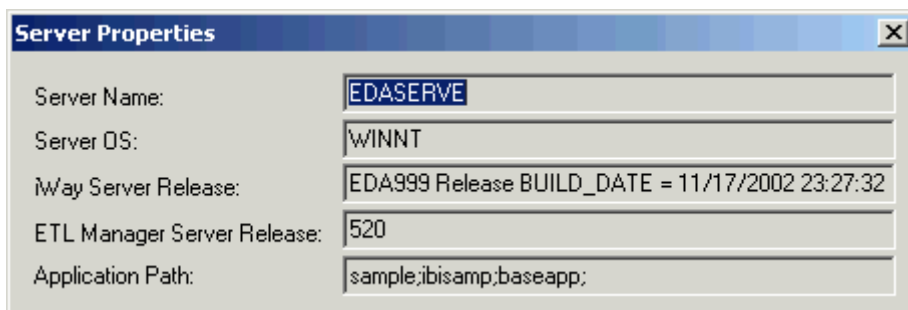
You can view the properties for a server with the Properties dialog box.

Procedure How to View a Server's Properties

Right-click the Server, and select *Properties* from the pop-up menu.

The Server Properties dialog box opens. For details on the Server Properties dialog box, see *Server Properties Dialog Box* on page 3-5.

Reference Server Properties Dialog Box



The Server Properties box contains the following fields/options:

Server Name

Displays the server name.

Server OS

Displays the operating system on which the server is running.

iWay Server Release

Is the release number date of the Server.

ETL Manager Server Release

Is the release number of the ETL Manager Server you are running.

Application Path

Displays the directories currently listed in APP PATH.

Using the Message Window

The ETL Manager Message Log window shows results from the ETL Workbench’s communications with the Server. There are two types of messages:





- **Error messages** inform you that you that you are unable to connect to a server.
- **Informational messages** inform you that requests were downloaded or saved.

By default, ETL Manager messages display. You can also set the message window to display additional messages. This is useful for trouble shooting and displaying other types of messages. For details, see *How to Display Additional Messages in the Message Window* on page 3-6.

Procedure **How to View the Message Window**

Select *Message Log* from the View menu.

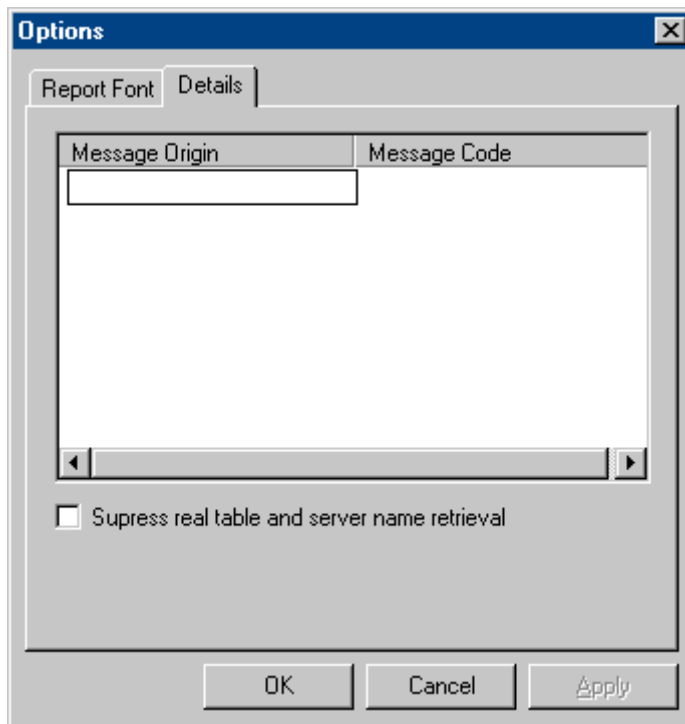
The Message Log window opens:

Date	Time	User	Server	Message
 07/01/1 ...	11:25:4...	doccn	CMINT	Connect successful
 07/01/1 ...	11:28:4...	doccn	CMINT	Download Copy request atest successful
 07/01/1 ...	11:29:0...	doccn	CMINT	Download Copy request testjh3 successful
 07/01/1 ...	11:29:3...	doccn	CMINT	Download Copy request ventest1 successful

Procedure **How to Display Additional Messages in the Message Window**

1. Choose *Options* from the Tools menu, and select the *Details* tab.

The Options window opens with the Details tab open:



2. Click under the Message Origin heading, and enter the message origin for the type of message you want to display. The message origin can be one of the following:
 - EDA* displays messages from the Server.
 - FOC* displays messages from the FOCUS language processor.
 - BLOB* displays second and subsequent message lines.
 - ** displays all types of messages.
3. Click under the Message Code heading and enter the message code for the type of message origin.
 - Entering a message number will display only that message.
 - Entering an asterisk displays all message with the specified origin.
4. Optionally, select *Supress real table and server name retrieval* to avoid information on the data source and server from being retrieved with error message information.

Ending an ETL Workbench Session

When you are finished working in ETL Manager, you can end your session.

Procedure How to End Your ETL Workbench Session

Select *Exit* from the File menu.

If you have any unsaved ETL Requests, a dialog box will display, asking if you want to save changes. For details on saving your ETL Request, see *Saving an ETL Request* in Chapter 7, *Working With an ETL Request*.

Getting Help

You can access the ETL Manager online help from within the ETL Workbench.

Procedure How to Access Online Help

Do one of the following:

- Click the *Help* button:



or

- Select *Help Topics* from the Help menu.

Information Builders on the Web

You can visit the Information Builders Web site where you can ask questions, get help, and look at Information Builders' new products and information. The Web site is located at <http://www.informationbuilders.com>.

For the latest information on ETL Manager, visit the iWay data warehouse technical support site located at <http://techsupport.ibi.com/etl.html>, or accessed by selecting *iWay on the Web* from the ETL Manager Help menu.

CHAPTER 4

Building an ETL Request

Topics:

- Beginning an ETL Request
- Using the Workflow Tool
- Identifying Data Sources
- Specifying Joins
- Specifying a Filter
- Selecting Columns
- Specifying a Sort
- Testing the SQL
- Specifying Transport Options
- Selecting a Data Target
- Specifying Mappings
- Specifying Data Target Options

This topic describes how to create and build an ETL Request.

Beginning an ETL Request


An ETL Request specifies the information to extract from the data sources and copy to the data target.

You can create a new ETL Request in ETL Manager and then build the request with the Workflow tool. For details on the Workflow tool see *Using the Workflow Tool* on page 4-6.

Creating a New ETL Request

The first step in building an ETL Request is to create a new one.

Procedure How to Create a New ETL Request

1. Do one of the following:
 - Select *New* from the File menu.
 - Right-click the *ETL Requests* folder, and select *New* from the pop-up menu.
 - Click the *New* button. 

The ETL Request opens.

2. If you are not connected to the selected server, the Enter Connection Information window opens so that you can connect. For details on connecting to a server, see *Connecting to the Server* in Chapter 3, *Getting Started*.

Setting Properties of an ETL Request

You can specify ETL Request properties in the Properties window. You can set options such as Scheduler settings, e-mail notification, and the application the ETL Request belongs to.

Procedure How to Set an ETL Request's Properties

1. If the Properties window is not open, click the *General* button.
2. In the Description field, enter a description of the ETL Request.
3. Select a folder from the Application drop-down list, or enter the name of a new folder. This is the folder in which the request will be saved.
4. Optionally, select *Run Stored Procedures Only (RPCs)* if you want the procedures associated with the ETL Request to run without executing the ETL Request.

- Optionally, select an option for e-mail notification in the Notify section. The options are:

Start sends an e-mail when an ETL Request is executed.

End sends an e-mail when an ETL Request's execution is successfully completed.

Failure of Job to sends an e-mail when a specified error occurs.

Note: An SMTP server is required to send e-mail notifications. For details on configuring an SMTP server, see Chapter 12, *Using the iWay Web Console*.

Reference Properties Window

Properties

Description

Application

Scheduler Status

☐ Run Stored Procedures Only (RPCs)

Created by: ck05271 Last Modified on (None) at (None)

Notify

Email at ☐ Start ☐ End ☐ Failure of Job to

Restart/Recovery

Number of Attempts

Restart from

The Properties window contains the following fields/options:

Properties

Contains the properties of the ETL Request.

Description

Is a description of the purpose of the ETL Request.

Application

Is the application in which the ETL Request resides.

Scheduler Status

Determines whether the Scheduler will start the ETL Request. The options are:

None indicates that a status has not been selected.

Active indicates that the Scheduler will execute the ETL Request when the Scheduler runs.

Inactive indicates that the ETL Request will not be executed when the ETL Request runs.

Run Stored Procedures Only (RPCs)

Specifies that only the procedures associated with the ETL Request will be run. The ETL Request will not be.

Created by:

Is the user who created the ETL Request.

Last Modified on

Is the date and time at which the ETL Request was last modified.

Notify

Determines notification of the results of the ETL Request.

Email at

Determines whether an e-mail will be sent at the start, end, or failure of an ETL Request execution. The options are

Start sends an e-mail when an ETL Request is executed.

End sends an e-mail when an ETL Request's execution is successfully completed.

Failure of Job to sends an e-mail when a specified error occurs.

If you do not want an e-mail sent, do not select any check boxes.

Note: An SMTP server is required to send e-mail notifications.

Restart/Recovery

Determines the behavior of ETL Manager when restarting an ETL Request.

Number of Attempts

Is the number of times to restart your request.

Restart From

Determines at what point in the ETL Request the request is restarted at. The options are:

Beginning restarts the request at the first record.

Last Commit restarts the request after the last commit point. This option is only effective for database target data. It is not useful for a flat file.

Building an ETL Request

Use the Workflow tool to build an ETL Request. Building an ETL Request consists of the following steps. Not all steps are necessary for every ETL Request.

1. Identify the sources. For details, see *Identifying Data Sources* on page 4-9.
 2. Specify any joins. For details, see *Specifying Joins* on page 4-17.
 3. Specify filters. For details, see *Specifying a Filter* on page 4-22.
 4. Select the columns from your data sources. For details, see *Selecting Columns* on page 4-30.
 5. Specify sorts. For details, see *Specifying a Sort* on page 4-39.
 6. Test the SQL. For details, see *Testing the SQL* on page 4-42.
 7. Specify FTP transport options. For details, see *Specifying Transport Options* on page 4-44.
 8. Specify a data target. For details, see *Selecting a Data Target* on page 4-46.
 9. Specify mappings. For details, see *Specifying Mappings* on page 4-56.
 10. Specify data target options. For details, see *Specifying Data Target Options* on page 4-77.
- For details on the Workflow tool, see *Using the Workflow Tool* on page 4-6.

Using the Workflow Tool

The Workflow tool is used to build your ETL Request. It consists of a workspace and a components palette.

- The Components palette contains each of the components that make up your ETL Request.
- The workspace allows you to arrange and join the components that make up your ETL Request.

Reference Components Palette



The Components palette contains the following components:

Source

Adds a data source to the ETL Request.

Join

Joins two or more data sources.

Filter

Applies filters to the data retrieval.

Column

Selects the columns in a data source used in the ETL Request.

Sort

Controls the order of the retrieved data.

SQLAnswerSet

Tests the SQL in the ETL Request.

Transport

Specifies the use of FTP Series to transfer your data.

Target

Specifies the data target and mappings.

Load

Specifies options for the data target.

Adding and Deleting a Component

You add components to an ETL Request using the components from the Components palette. If a component is no longer needed, you can eliminate it from your request by deleting it.

Procedure How to Add a Component to Your ETL Request

Click and drag the component from the Components palette to the workspace.

Procedure How to Delete a Component From Your ETL Request

Right-click the component you want to remove, and select *Delete* from the pop-up menu.

Connecting and Disconnecting Components in Your ETL Request

Connecting a component to another component in the workspace allows information to flow between the two components. You connect and disconnect components using ports, which are the gray objects located on the upper left and lower right of each component.

Procedure How to Connect Components

1. Click a port on the one of the components and hold the mouse button down.
Your cursor turns into a pen.
2. Drag the pen to a port on the component you want to connect to, and release the mouse button.

Procedure How to Disconnect Components

Right-click a port on one of the components you wish to disconnect, and select *Delete connections* from the pop-up menu.

Opening and Closing a Component

After adding a component to an ETL Request, you can specify the behavior of each component by opening it and using the fields and options in the window that opens.

Procedure How to Open a Component

With the component in the workspace:

Double-click the component.

or

Right-click the component and select *Open* from the pop-up menu.

Procedure How to Close a Component

Right-click the title bar and select *Close*.

or

Double-click the control icon on the title bar.

Procedure How to Maximize a Component

Right-click the component and select *Maximize* from the pop-up menu.

Selecting Fields in a Component

When working in a component, there are several ways to select the fields you want to use in your ETL Request.

Procedure How to Select a Single Field

Click the gray box to the left of the field.

Procedure How to Select Multiple Fields

Hold the *Ctrl* key and click the gray box next to each field you want to select.

or

To select contiguous fields, click the gray box next to the first field, hold down the *Shift* button, and select the last field you want included.

Procedure How to Select All Fields

Click the gray box at the top left of the list.

Identifying Data Sources

The first step in building an ETL Request is to identify the data sources.

When creating an ETL Request, information about the data source such as the column names is required, and is supplied when ETL Manager queries the iWay Catalog.

When creating or editing an ETL Request, it is required that any data sources referenced in the ETL Request are available:

- A relational data source must be running for data in that data source.
- A data set must be allocated for data in a flat file or VSAM structure.
- The source servers must be running and be reachable from the Hub server in a typical configuration with a Hub server and one or more source servers.

Procedure How to Identify a Data Source for Your ETL Request

1. Add a Source component to the workspace.
2. Double-click the Source component to open it. The Source window opens with a list of available data sources. For details on the Source window, see *Source Window* on page 4-10.
3. Scroll through the list of data sources to find the one you want to use, and select it by double-clicking the gray box to the left of the data source, or selecting the data source and clicking *Select Table*.

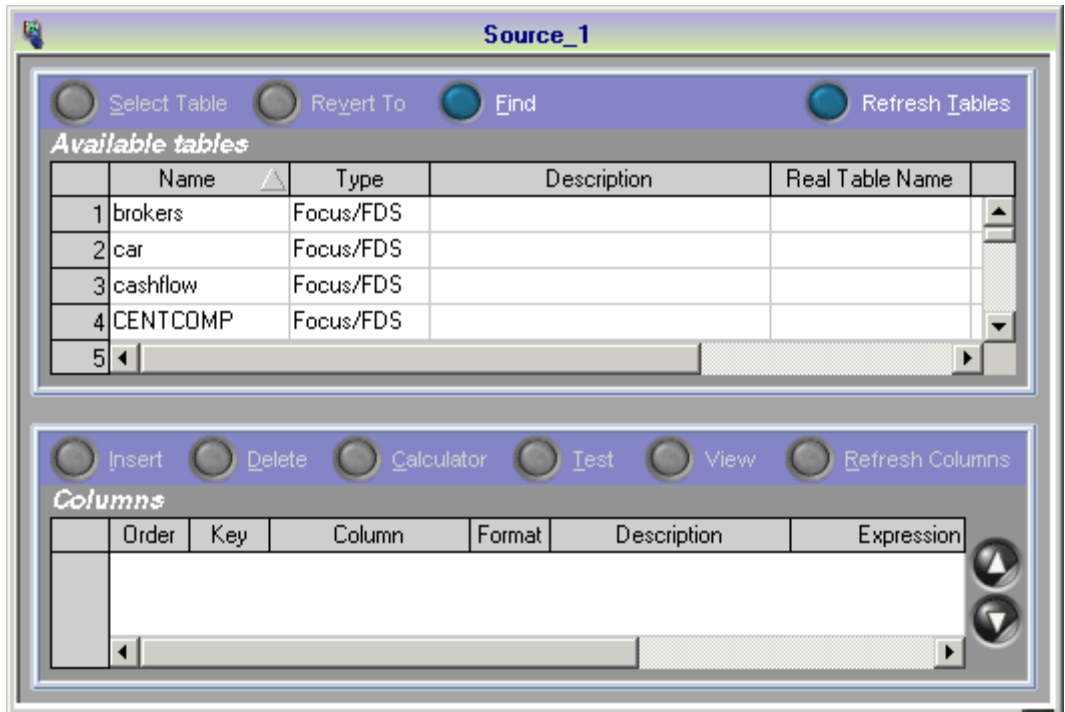
A list of the columns from the selected data source appears in the Columns grid, and the Available tables grid is hidden.

Note: You can preview the columns in a data source by clicking the gray box next to the data source name. Double-clicking will select it.

Optionally, create an extract transform. For details, see *Creating an Extract Transform* on page 4-13.

4. Double-click the control icon on the title bar to close the Source window.

Reference Source Window



The Source window contains the following fields/options:

Select Table

Selects the highlighted table.

Revert to

Reverts the window to the previous selection.

Find

Allows you to search for a data source by name, type, or character string.

Refresh Tables

Refreshes the list of data sources displayed in the Available tables grid.

Available tables

Displays the data sources available on the Server you are connected to.

Name

Is the Synonym name.

Type

Is the data source type.

Description

Is the description of the data source.

Real table name

For relational data sources, is the name of the data source.

Connection

For data sources with multiple connectors, is the name of the connection with which to access the data.

Note: If you double-click the Name, Type, or Description heading the data sources are sorted by that field.

Insert

Inserts an extract transform in the Columns grid.

Delete

Deletes an extract transform from the Columns grid.

Calculator

Opens the Extract Transform calculator.

Test

Tests extract transforms.

Refresh Columns

Refreshes the list of columns in the selected data source.

View

Displays a sample of rows from the selected data.

Up and Down arrows

Moves the selected extract transform up or down in the Columns grid.

Columns

Displays the list of columns contained within the selected data source.

Order

Is the order in which the columns appear in the Synonym.

Key

Displays a symbol that indicates whether the column is a column in the data source, a key column, or an extract transform.



Is a key column.



Is a column.



Is a virtual column described in an extract transform.



Is a virtual column described in the file description.



Is an indexed column. This is only available in a FOCUS or Fusion data source.

Column

Is the column name.

Format

Is the format of the column. For details on formats, see *Field Formats and Mapping* on page 4-72.

Description

Is the description of the column, if available.

Expression

For extract transforms only. Is the definition of the extract transform.

Location

If the data source is a relational data source, specifies an extract transform should be performed on the local Hub Server of the remote subserver.

Error

Describes any errors in an extract transform.

Updating the List of Data Sources

You can refresh the list of available data sources, and the list of columns in a data source. This is useful when you add data sources or columns to the iWay Catalog while building an ETL Request, and want these changes reflected in the data source or column list.

Procedure How to Update the List of Data Sources

In the Source window, click *Refresh Tables*.

Procedure How to Update the List of Columns

In the Source window, click *Refresh Columns*.

Creating an Extract Transform

You can create an extract transform in a data source to add intermediate calculations to the request. An extract transform is performed as the records are read, and before selection filters are performed. This is often useful for complex calculations that are composed of multiple expressions. Creating an extract transform makes the transformation easier to understand and support. It is also helpful in performing data type conversions when multiple steps are required.

An extract transform can contain an expression or a constant.

- If the extract transform contains an expression, you can create the expression in the Expression field or with the Calculator.
- If the extract transform contains a constant value, you can enter the value in the Expression field. If this is the first extract transform, you must identify the segment the new column should be associated with.

ETL Manager draws a black line in the list of columns to designate where extract transforms are created. Extract transforms that display above the line are created in the Master File. Extract transforms that display below the line are created in the ETL Request.

When an extract transform is created in the Workflow, it is immediately parsed. If there is an error, the extract transform displays in red type.

After creating an extract transform, you can test it from the Source window. For details, see *How to Test an Extract Transform* on page 4-15.

Procedure How to Create an Extract Transform in the Columns Grid

1. In the Source window, click *Insert*.
A new column is added at the end of the list of columns below a black line.
2. Double-click in the Column field and enter a column name.
3. Double-click the Format field and enter a format.
4. Optionally, double-click the Description field and enter a description for the column.
5. Double-click in the Expression field, and enter the expression for the extract transform.

Procedure How to Create an Extract Transform With the Calculator

1. In the Source window, click *Insert*.
A new column is added at the end of the list of columns below a black line.
2. Click *Calculator*.
The Extract Transform calculator opens.
3. Enter a column name in the Column field.
4. Enter a format in the Format field, or click *Format* to display the Format dialog box which allows you to select a format.
5. Optionally, select *Allow lower case format* to allow the format to be entered in mixed or lower case.
6. Enter the calculation you want to perform using the calculator buttons, Columns tab, and Functions tab, then click *OK*. For details on using the calculator, see *Extract Transform Calculator* on page 4-15.
7. Optionally, double-click the Description field and enter a description for the column.

Procedure How to Create a Counter With an Extract Transform

1. In the Source window, click *Insert*.
A new column is added at the end of the list of columns below a black line.
2. Double-click in the Column field and enter a name for the counter, for example, *CNTR*.
3. Double-click the Format field and enter a format.

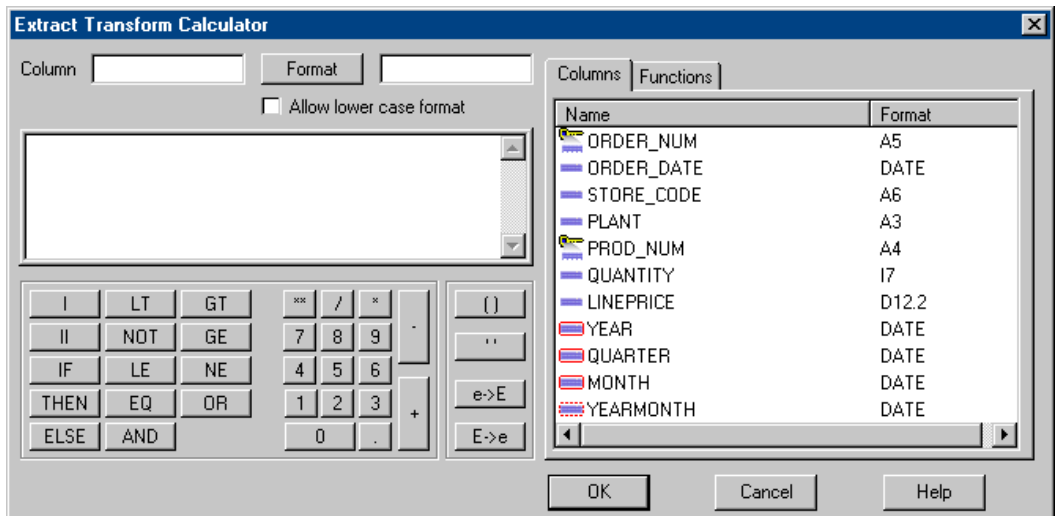
Note: For a multi-segment FOCUS or FUSION file, the segment that will contain the counter must be specified by adding a WITH phrase in the Format column.

4. Optionally, double-click the Description field and enter a description for the column.
5. Double-click in the Expression field, and enter CNTR + n where n is the number you want to count by.

Procedure How to Test an Extract Transform

Click *Test*.

Reference Extract Transform Calculator



The Extract Transform calculator contains the following fields/options:

Column

Is the column name.

Format

Is the column format.

Allow lower case format

Allows the format to be entered in mixed or lower case.

Expressions window

Displays the expressions you create.

Columns

Displays a list of available columns.

Functions

Displays a list of functions that are available for your calculations. For details on functions, see *Using Functions* on page 4-65.

Conditional buttons

Inserts conditions into the expressions window.

Numeric buttons

Inserts numbers and numeric operators into the expressions window.

()

Adds parentheses in the calculation.

"

Inserts two single quotation marks into the expressions window. Enter alphanumeric test values between these.

e->E

Converts selected text in the expressions window to uppercase.

E->e

Converts selected text in the expressions window to lowercase.

OK

Accepts your calculation, and adds it to the column.

Cancel

Cancels the calculation.

Help

Accesses the online help system.

Specifying Joins

If you select more than one data source for your ETL Request, you must join them using the Join component. This is done by specifying a common column or columns that will link the tables to each other.

You can perform an inner, left outer, right outer, or cross join.

- An *inner join* extracts those rows that appear in both tables.
- A *left outer join* includes all of the rows from the table listed in the Left source columns grid and the columns from the Right source columns grid that match one of these columns.
- A *right outer join* includes all of the rows from the table listed in the Right source columns grid and the columns from the Left source columns grid that match one of these columns.
- A *cross join* is a Cartesian product of two tables. It consists of all possible pairs of rows from the two tables.

Note: You can create a recursive join by selecting the same data source twice and joining them.

When you link data sources, ETL Manager identifies each table with T_n , where n is the order in which the table was chosen. For example, the first source chosen would be identified as T_1 .

In ETL Manager, join syntax is generated in ANSI 92 SQL syntax. For example, when selecting the tables Product and Vendor, and columns VENDORNUM and VENDOR_NUM, ETL Manager generates:

```
SELECT...FROM
Product t1 INNER JOIN VENDORS T2 on t1.vendornum=T2.vendor_num
```

Previous versions of ETL Workbench constructed a join with a WHERE criteria, and the following syntax was generated:

```
SELECT...FROM
product T1, vendors T2
WHERE T1.vendornum=T2.vendor_num
```

If you prefer to use a WHERE criteria, select a cross join as the join type and build a WHERE criteria using the Filter component.

Reference Considerations for Join Performance

iWay lets you create joins across similar data sources, different data sources, and servers (also known as *cross-platform joins*). However, you cannot join a comma delimited file with a flat file.

In general, cross-platform joins and joins without key fields can create many intermediate files, depending on the number of tables joined. This issue may be less significant for some iWay platforms. For example, the Server for OS/390 can use hyperspaces and alternate sort utilities. If performance is a concern, you should carefully evaluate the impact of your joins.

To improve performance, try any of the following:

- Minimize the number of tables used in a join.
- Join tables using key columns and indexes, when possible.
- Make sure that the *smaller* of the two tables is selected first and the *larger* one is selected second.

Procedure How to Join Data Sources

1. Add a Join component to the workspace.
2. Connect the Join component to the data sources you are joining.
3. Double-click the Join component to open it. The Join window opens, listing the columns from one data source in the Left source columns grid and the columns from the other data source in the Right source columns grid.

For details on the Join window, see *Join Window* on page 4-20.

4. Select the type of join you want by clicking the join diagram.
 - Click the left circle for a **left outer join**. A left outer join includes all of the rows from the table listed in the Left source columns grid and the columns from the Right source columns grid that match one of these columns.
 - Click the right circle for a **right outer join**. A right outer join includes all of the rows from the table listed in the Right source columns grid and the columns from the Left source columns grid that match one of these columns.
 - Do not click either circle for an **inner join**. An inner join extracts those rows for vendors who appear in *both* tables. This is the default.
 - Double-click the selected join for a **cross join** or **Cartesian product**.

5. Create the join:

- a.** Select a field from the Left source columns grid.
- b.** Select the matching field from the Right source columns grid.

- c.** Click .

The condition appears in the Join conditions grid.

Note: If you add incorrect fields to the Join conditions grid, you can delete the condition by selecting it and clicking *Delete*.

6. Optionally, add a calculation or function to the condition:

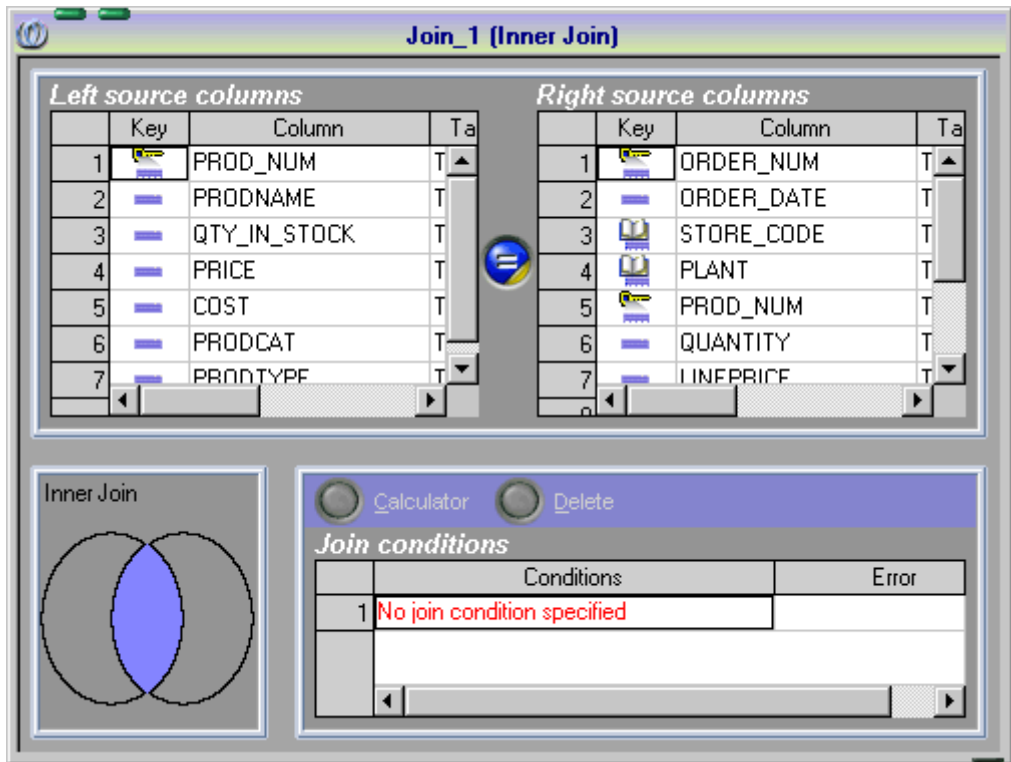
- Click the condition to select it, then click it again to change the text. Then add a calculation or function in the Expressions window. For details on functions, see *Using an SQL Function* on page 4-37.

or

- Use the Join Calculator to create a calculation. For details, see *Join Calculator* on page 4-21.

7. Double-click the control icon on the title bar to close the Join window.

Reference Join Window



The Join window contains the following fields/options:

Left source columns

Displays the available columns from the data source connected to the left join port.

Right source columns

Displays the available columns from the data source connected to the right join port.

=

Establishes the relationship between the selected columns as equal.

Calculator

Opens the Join Calculator.

Delete

Deletes the selected join condition.

Join conditions

Lists the conditions by which the two data sources are connected.

Join diagram

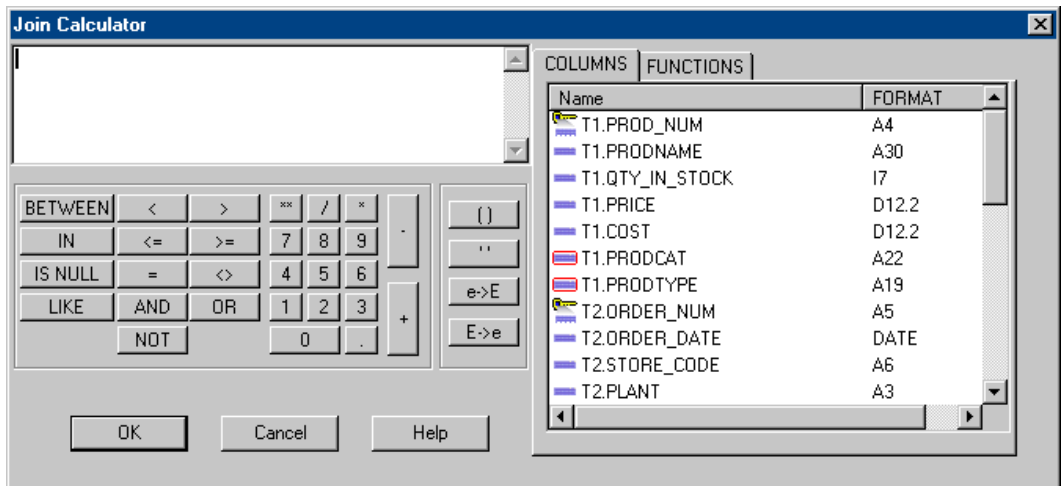
Determines the type of join being used.

Procedure How to Create a Join Condition With the Join Calculator

1. Select the condition, and click *Calculator*.

The Join Calculator opens.

2. Enter your calculation in the Expressions box by typing it or using the *FUNCTIONS* and *COLUMNS* tabs, and the calculator buttons.
3. Click *OK*.

Reference Join Calculator

The Join Calculator contains the following fields/options:

Expressions window

Displays the expressions you create.

COLUMNS

Displays a list of available columns.

FUNCTIONS

Displays a list of functions that are available for your calculations. For details on functions, see *Using Functions* on page 4-65.

Conditional buttons

Inserts conditions into the expressions window.

Numeric buttons

Inserts numbers and numeric operators into the expressions window.

()

Adds parentheses in the calculation.

"

Inserts two single quotation marks into the expressions window. Enter alphanumeric test values between these.

e->E

Converts selected text in the expressions window to uppercase.

E->e

Converts selected text in the expressions window to lowercase.

OK

Accepts your calculation, and adds it to the column.

Cancel

Cancels the calculation.

Help

Accesses the online help system.

Specifying a Filter

You can create a filter to control the data included in your ETL Request. A filter creates WHERE criteria in the SQL of your ETL Request. You can filter your data with a constant value or a variable.

Procedure **How to Add a Filter**

1. Add a Filter component to the workspace.
2. Connect the Filter component to the Source component, or to the Join component if you are using one.
3. Double-click the Filter component to open it. The Filter window opens. For details, see *Filter Window* on page 4-24.
4. Select the field in the All columns grid that you want to filter the data with.

5. Enter an expression in the Condition field. This can be a field name, or one of the following:
 - A **date** must be in the format 'yyyy/mm/dd'.
 - A **number**.
 - An **alphanumeric value** must be enclosed in single quotation marks.
 - A **variable**. If you add a variable, a Parameters Setup window will prompt you for default values when you test or verify the query. For details on using a variable in a filter see *Using a Variable in a Filter* on page 4-28.
6. Double-click the field you are filtering data with.

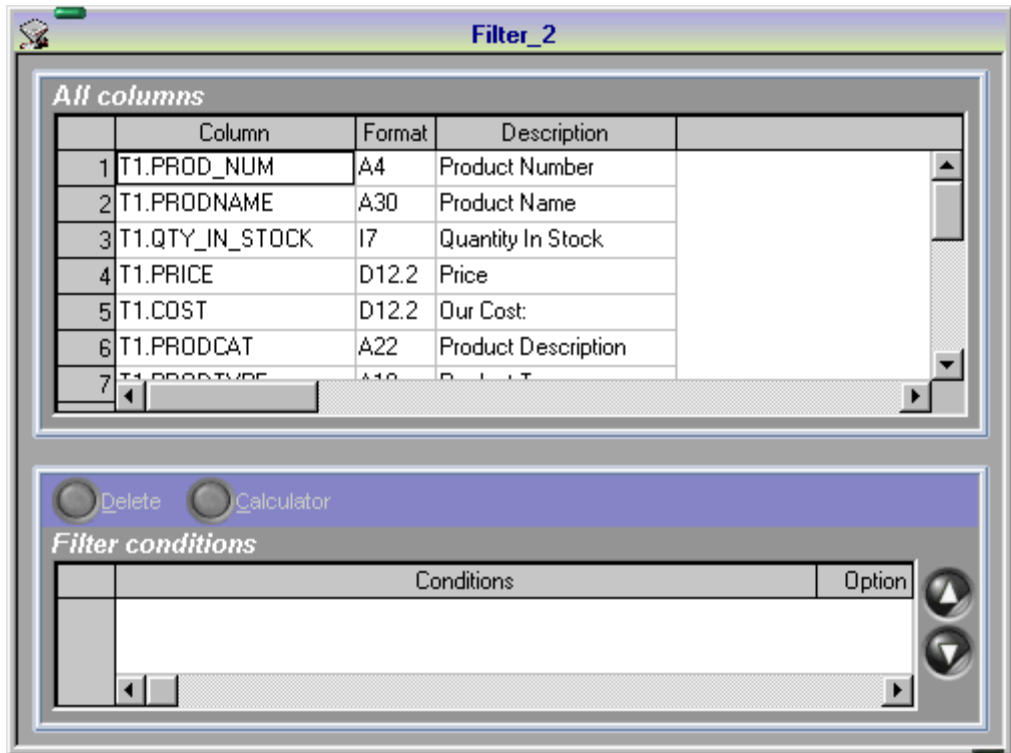
The filter appears in the Filter conditions grid.
7. Optionally, create a more complex filter condition:
 - a. Double-click the condition.

The Edit Condition calculator opens.
 - b. Use the calculator to create a condition. For details on the Edit Condition calculator see *Edit Condition Calculator* on page 4-25.
 - c. Click OK.
8. Optionally, repeat steps 4 through 7 to add more filters.
9. If you add additional filters, specify whether all or one of the filters need to be satisfied for the data to be included by double-clicking in the Option field and selecting a value from the drop-down list.

AND includes data if all conditions are met.

OR includes data if one of the conditions is met.
10. Double-click the control icon on the title bar to close the Filter window.

Reference Filter Window



The Filter window contains the following fields/options:

All columns

Lists the available columns.

Column

Is the column name.

Format

Is the format of the column.

Description

Is the description of the column, if available.

Operation

Is the relation being used for the filter.

Condition

Is the value that filters the data.

Delete

Deletes the selected condition from the Filter conditions grid.

Calculator

Opens the Filter calculator which allows you to create complex filters.

Filter conditions

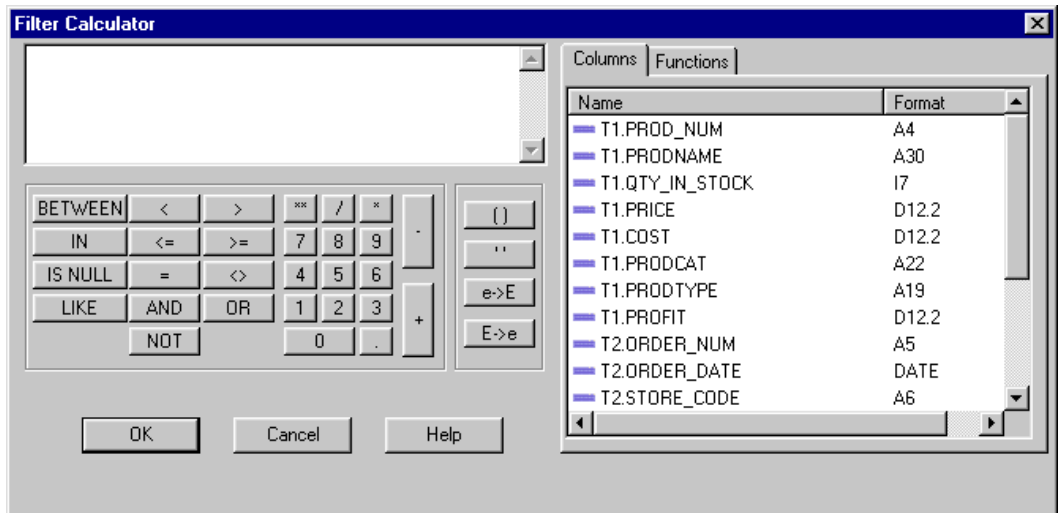
Displays the conditions to be applied to your data.

Conditions

Displays the condition to be applied to your data.

Option

If there is more than one condition, specifies whether all or only one need to be satisfied for the data to be included.

Reference Edit Condition Calculator

The Edit Condition calculator contains the following fields/options:

Expressions window

Displays the expressions you create.

BETWEEN

Specifies data will be returned if the value of the field's value falls between the values entered on either side of the word AND. For example:

`RETAIL BETWEEN 20 AND 30`

IN

For alphanumeric data, specifies data will be returned if the field's value is a substring of the right operand.

IS NULL

Specifies that records with a null value for the specified column are returned.

LIKE

Returns data if the data in the field's value matches the character string on the right operand. In the character string, use the underscore character (_) to represent any single character, and the percent character (%) to represent any sequence of characters. For example

```
LAST_NAME LIKE '%SMITH%'
```

returns the value TRUE if LAST_NAME contains the string SMITH anywhere inside it (for example, SMITH, SMITHSON, SILVERSMITH).

```
LAST_NAME LIKE 'S _ _ _ _'
```

returns the value TRUE if LAST_NAME is exactly 5 characters long and the first letter is S.

<

Returns data if the field's value is less than the right operand. For example

```
SALARY < 30000
```

restricts the query to those employees who earn less than \$30,000.

<=

Returns if the field's value is less than or equal to the right operand.

=

Specifies data if the field's value equals the right operand.

AND

Is used in combination with the BETWEEN operator.

NOT

Is used in combination with other relational operators to negate them. For example

```
LAST_NAME IS NOT NULL
```

returns all records where the value of LAST_NAME is not null.

>

Specifies data will be returned if the field's value is greater than the right operand.

>=

Specifies data will be returned if the field's value is greater than or equal to the right operand.

<>

Specifies data will be returned if the left operand is not equal to the right operand.

OR

Inserts the operator OR into the expressions window.

Raises a value to the specified power.

0-9

Inserts the number into the expressions window.

/

Divides values.

Multiplies values.

.

Inserts a decimal point into the expressions window.

-

Subtracts a value.

+

Adds values.

()

Inserts parentheses in the expressions window. These are used to determine the order in which ETL Manager performs calculations.

''

Inserts two single quotation marks into the expressions window. Enter alphanumeric test values between these.

e->E

Converts selected text in the expressions window to uppercase.

E->e

Converts selected text in the expressions window to lowercase.

Columns

Displays all available columns.

Functions

Displays a list of available functions to use in your calculation. For details, see *Using an SQL Function* on page 4-37.

Creating a Sub-Query in a Filter Condition

Although ETL Manager does not generate sub-queries, they can be typed into a screening condition.

Example Using a Sub-query in a Filter Condition

If you wanted all products from a vendor that offered a cash discount greater than five percent, create a WHERE criteria:

```
VENDORNUM IN (SELECT VENDOR_NUM FROM VENDORS WHERE CASH_DISC >5)
```

Using a Variable in a Filter

You can use a variable to filter data in your ETL Request. At run time, the Server executes the SQL command from a procedure, which permits the use of variables in a filter condition.

A variable is identified by a double ampersand and an alphanumeric name, up to ten characters long. For example:

- &&DEPTNO
- &&CURDATE
- &&JOBID

Note: Do not use names starting with &&CM_; these are used internally by the Server.

When you run an ETL Request that includes the values saved in the Parameters Setup window, these values are used as *default* values. They can be overridden at run time in a user profile or with the CMOPTION procedure specified as a pre-extract procedure. For information, see Chapter 8, *Controlling ETL Requests With Procedures*.

Procedure How to Use a Variable in a Condition

1. Select the condition in the Filter conditions grid and click *Calculator*.
The Edit Condition calculator opens.
2. Enter your condition using a variable for the value. If the column contains character or date values, enclose the variable name in single quotation marks.
The Parameter Setup window opens.
3. Enter a default value in the Value field for the variable, and click OK.

Example Using a Variable With a Numeric Value in a Condition

If you enter the following filter condition

`T1.QTY_IN_STOCK = &&INSTOCK`

the Parameters Setup window opens, prompting you for a default value for &&INSTOCK.

	Parameters	Value
1	INSTOCK	

OK Cancel

Example Using a Variable With a Date Value in a Condition

If you enter the following filter condition

`T2.ORDER_DATE= '&&ODATE'`

the Parameters Setup window opens, prompting you for a default value for &&ODATE.

Example Using a Variable With an Alphanumeric Value in a Condition

If you enter the following filter condition

```
T2 . PRODUCT= ' &&CAT '
```

the Parameters Setup window opens, prompting you for a default value for &&CAT.

Selecting Columns

The columns you select define the input to your data target. If you are using ETL Manager to format a new data target, the order of the columns will reflect the order that the columns are specified in the SQL command.

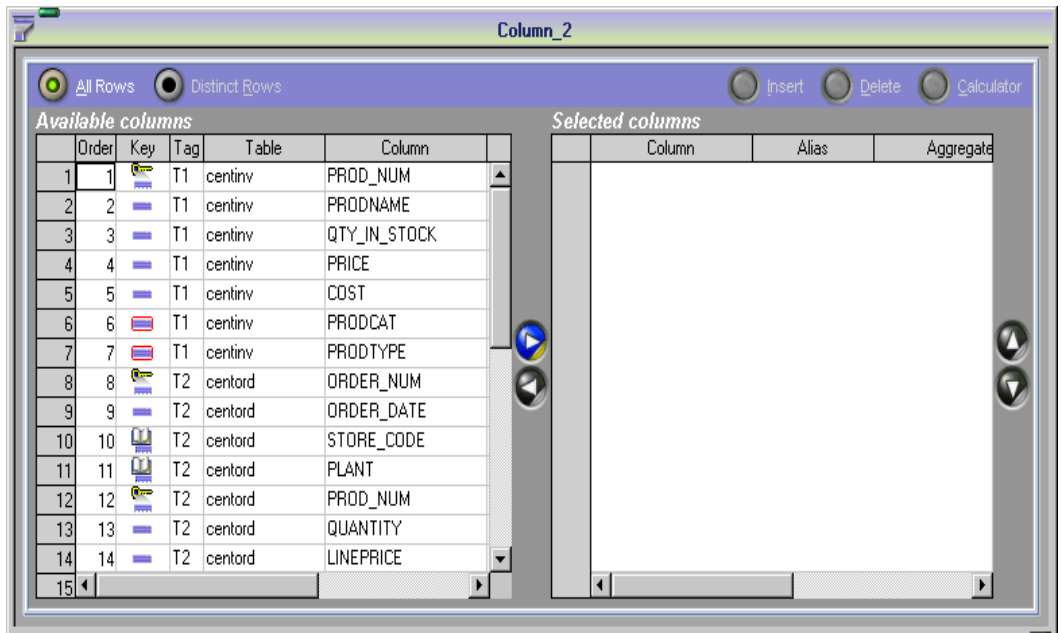
The Server, which ETL Manager uses to process its extraction logic, can have up to 400 columns in a single request. If you intend to move more than 400 columns, several requests may be linked together by creating dependencies. For information on dependencies see Chapter 5, *Scheduling and Executing ETL Requests*.

Procedure How to Select Columns

1. Add a Column component to the workspace.
2. Connect the Column component to the Source component, or the Join or Filter component if you are using one.
3. Double-click the Column component to open it. The Column window opens. For details on the Column window see *Column Window* on page 4-31.
4. Select the columns you want to use from the Available columns grid, and click the right arrow. The selected columns display in the Selected columns grid.
5. Optionally, rename the selected columns by double-clicking in the Alias field and entering a name.

An alias name must start with a letter; can contain a combination of letters, numbers, and underscores; and must be enclosed in double quotation marks.
6. Put the selected columns in the order you wish by selecting the columns and clicking the up and down arrows.
7. Optionally, create an SQL calculation. For details, see *Creating an SQL Calculation* on page 4-34.
8. Optionally, aggregate the columns. For details, see *Aggregating Columns* on page 4-34.
9. Optionally, click Select *Distinct* to remove duplicate rows from the results of your query.
10. Double-click the control icon on the title bar to close the Column window.

Reference Column Window



The Column window contains the following fields/options:

All rows

Selects all columns in the Available columns grid.

Distinct Rows

Removes duplicate rows from the results of your query.

Insert

Inserts a new column.

Delete

Deletes the selected column.

Calculator

Opens the SQL calculator. For details on the SQL calculator see *SQL Calculator* on page 4-36.

Available columns

Displays the available columns from your data sources.

Order

Is the order in which the columns display.

Key

Displays a symbol that indicates whether the column is a column in the data source, a key column, or an extract transform.



Is a key column.



Is a column.



Is a virtual column described in an extract transform.



Is a virtual column described in the file description.



Is an indexed column. This is only available in a FOCUS or Fusion data source.

Tag

Is the alias or table number of the data source. The number assigned to the Tag field is determined by the order in which the data source was selected.

Table

Is the name or alias that identifies the data source (for example, T1 or T2).

Column

Is the column name.

Format

Is the column format.

Description

Is the description of the column, if available.

Error

Describes any error that may exist in the column or the column's expression. For details, see *Column Parsing Errors* on page 4-33.

Double-click a heading to sort the columns by that field.

Selected columns

Displays the columns chosen to be in the ETL Request.

Column

Is the column name, or an expression created in the SQL calculator.

Alias

Is an alternate name for the column.

Aggregate

Is the aggregation applied to the column. For details, see *Aggregating Columns* on page 4-34.

Description

Is an optional user-supplied description of the column.

Right arrow

Adds the selected column or columns to the Selected columns grid.

Left arrow

Removes the selected column or columns from the Selected columns grid.

Up and down arrows

Moves the selected column up or down within the list of selected columns.

Reference Column Parsing Errors

Columns and expressions are parsed by the Column component. If an error occurs, the columns that contain an error are turned red as a warning, and a message is displayed in the Error column in the Column window. The following error conditions may occur.

- **Error parsing the column expression** occurs because a calculation contains a syntax error.
- **One or more column is not found in available columns** occurs when a calculation uses a column name that does not exist in the selected source.
- **Duplicate columns will need to be aliased** occurs because the same column is selected twice in a SELECT command, but neither column has an alias. ETL Manager requires that all columns in a SELECT command have unique names. If the same column is selected twice, even with different aggregates such as MIN(COST) and MAX(COST) they must have different names. Enter a name under the ALIAS column for at least one of the columns
- **Select an aggregate for this column** occurs when some, but not all, columns are aggregated. SQL syntax requires that if any column is aggregated, they all must. Select an aggregation from the pull-down list in the Aggregate column.

Aggregating Columns

You can aggregate your columns in the Column window.

Procedure How to Aggregate a Column

1. Select the column or columns you want to aggregate. To select more than one column, hold the Ctrl button while selecting columns.
2. Double-click in the Aggregate field to display the drop-down list containing the types of aggregation, and select an aggregation. Your options are:

GroupBy groups the data by each distinct value in the column.

GroupBy Not Selected groups a column that you don't want to appear in the results.

Sum adds the values within the column.

Min returns the minimum value found in the column.

Max returns the maximum value found in the column.

Count returns a count of all non-null column values.

Avg returns the average value for the column.

Note: If you apply an aggregate function to one of your selected columns, the rest of the selected columns generally must have a function applied or be a GroupBy column. After an aggregation is applied to a field, all fields without an aggregation turn red until an aggregation is applied to them as well.

Creating an SQL Calculation

You can create an SQL calculation for your ETL Request that:

- Performs a calculation. For details, see *How to Create a Calculation* on page 4-34.
- Designates a constant value for a column. For details, see *How to Create a Constant Value for an SQL Calculation* on page 4-35.
- Adds date and time values. For details, see *How to Create a Time or Date Value as an SQL Calculation* on page 4-35.

Procedure How to Create a Calculation

1. In the Column window, click *Insert*.

A new column is added to the Selected columns grid.

2. Click *Calculator*.

The SQL calculator opens.

3. Enter the calculation you want to perform with the calculator buttons, Columns tab, and Functions tab. For details on using the calculator, see *SQL Calculator* on page 4-36.
 4. Optionally, enter an alias for the column in the Alias field.
 5. Click OK.
 6. The calculation appears in the Column field.
- Note:** You can use an SQL function in your extract transform by clicking the Functions tab in the SQL Calculator. For details on these functions, see *Using an SQL Function* on page 4-37.
7. Optionally, select an aggregation for the column in the Aggregate field. For details, see *Aggregating Columns* on page 4-34.

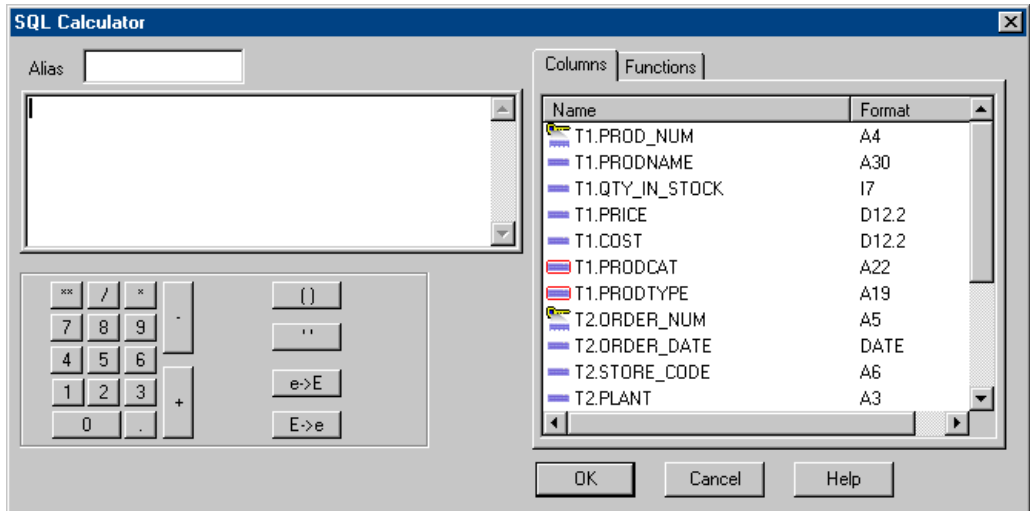
Procedure How to Create a Constant Value for an SQL Calculation

1. In the Column window, click Insert.
A new column is added to the Selected columns grid.
2. In the Column field, enter the value you want assigned to the column. Character and date values must be enclosed in single quotation marks.
3. In the Alias field, enter a name for the column.
4. Optionally, select an aggregation for the column in the Aggregate field. For details on aggregating columns, see *Aggregating Columns* on page 4-34.

Procedure How to Create a Time or Date Value as an SQL Calculation

1. In the Column window, click Insert.
A new column is added to the list of columns.
2. In the Column field, enter the one of the following:
CURRENT DATE adds the date.
CURRENT TIME adds the time.
CURRENT TIMESTAMP adds a timestamp.
3. In the Alias field, enter a name for the column.

Reference SQL Calculator



The SQL calculator contains the following fields/options:

Alias

Is an alias for the column.

Expressions window

Displays the expressions you create.

Columns

Displays a list of available columns.

Functions

Displays a list of SQL functions that are available for your calculations. For details, see *Using an SQL Function* on page 4-37.

Numeric buttons

Display the numbers and calculations available.

()

Adds parentheses in the calculation.

"

Inserts two single quotation marks into the expressions window. Enter alphanumeric test values between these.

e->E

Converts selected text in the expressions window to uppercase.

E->e

Converts selected text in the expressions window to lowercase.

OK

Accepts your calculation, and adds it to the column.

Cancel

Cancels the calculation.

Help

Accesses the online help system.

Reference SQL Operators

The following table describes the result of an operator on an expression or expressions, where *expr* is an expression.

Operator	Result
<i>expr</i> + <i>expr</i>	Returns the sum of two expressions
<i>expr</i> - <i>expr</i>	Returns the difference of two expressions
- <i>expr</i>	Returns the negated value of an expression
<i>expr</i> * <i>expr</i>	Returns the product of two expressions
<i>expr</i> / <i>expr</i>	Returns the dividend of two expressions
<i>expr</i> <i>expr</i>	Returns the concatenation of two character strings

Using an SQL Function

You can use an SQL function in the SQL calculator to create a complex expression. In addition, most ANSI 92 SQL functions can be used in a SELECT command, depending on the data source type and platform. For a list of SQL functions, see *SQL Functions and Descriptions* on page 4-38.

Procedure How to Use an SQL Function

1. In the SQL calculator, click the *Functions* tab.
2. Double-click the function you want to use.
The function appears in the expressions window.
3. Replace the placeholders with necessary values and click *OK*.

Reference SQL Functions and Descriptions

SQL Function	Description
<code>abs (expr)</code>	Returns the absolute value of <i>expr</i> .
<code>char (number)</code>	Converts a number to a character string.
<code>char_length (string)</code>	Returns the length of <i>string</i> in a character string.
<code>date (expr)</code>	Returns a date, time or timestamp value that is computed from <i>expr</i> .
<code>day (date)</code>	Returns a number that contains the day of the month from a date or timestamp.
<code>days (date)</code>	Returns the number of days since December 31, 1900.
<code>decimal (number [,precision [,scale]])</code>	Converts a number into a decimal value where <i>precision</i> is the number of digits in the result, and <i>scale</i> is the scale of the result.
<code>extract (field from expr)</code>	Extracts a numeric value from a date or time value <i>expr</i> , where <i>field</i> is either a year, month, day, hour, minute, or second.
<code>float (number)</code>	Converts <i>number</i> to a floating point value.
<code>integer ({number string})</code>	Converts a character string or number to an integer value.
<code>length (expr)</code>	Returns the number of bytes of storage used by <i>expr</i> .
<code>lower (string)</code>	Converts all the uppercase characters in <i>string</i> to lowercase.
<code>month(date)</code>	Returns the number of the month in which <i>date</i> falls.
<code>position (search in string)</code>	Returns the location of <i>search</i> in <i>string</i> , or returns a zero if <i>search</i> is not found.
<code>Substr (string, start, length)</code>	Returns a string from <i>string</i> starting at <i>start</i> to the end of the string, or if specified <i>length</i> characters.
<code>substring (string from start [for length])</code>	Returns a string from <i>string</i> starting at <i>start</i> to the end of the string, or if specified <i>length</i> characters.

SQL Function	Description
<code>trim</code> (<code>[Leading Trailing Both]</code> <code>[char] [From] string</code>)	Removes leading, trailing, or both characters <i>char</i> (or blanks if omitted) from a string.
<code>upper (string)</code>	Converts all lowercase characters in <i>string</i> to uppercase.
<code>value (expr)</code>	Returns the first non null value from <i>expr</i> . If all the values are null, returns null.
<code>year(date)</code>	Returns the year in which <i>date</i> falls.

Specifying a Sort

You can specify a sort to determine the order in which new columns appear in the results. A sort orders your results according to the results of a column you choose. You can sort by more than one column. When you do this, the sort field highest on your list sets the major sort break, the second sets the second sort break, and so on.

A sort is useful only in some cases. Use the following to consider whether or not to include a sort in your ETL Request.

- **Relational data target.** A sort is generally not useful because they are not respected when loading a relational data target.
- **FOCUS or Fusion data target.** A sort is useful for speeding the process of loading the data target. Loading without a sort will cause the process to take more time, and you will receive a warning message.
- **Flat file.** A sort is useful if you are loading your data into an application that respects sorting. If it does not, a sort is not useful with a flat file.

If the Sort component is used, it must be connected to the Column component.

Procedure **How to Specify a Sort**

1. Add a Sort component to the workspace.
2. Connect the Sort component to the Column component.
3. Double-click the Sort component to open it.

The Sort window opens. For details, see *Sort Window* on page 4-41.

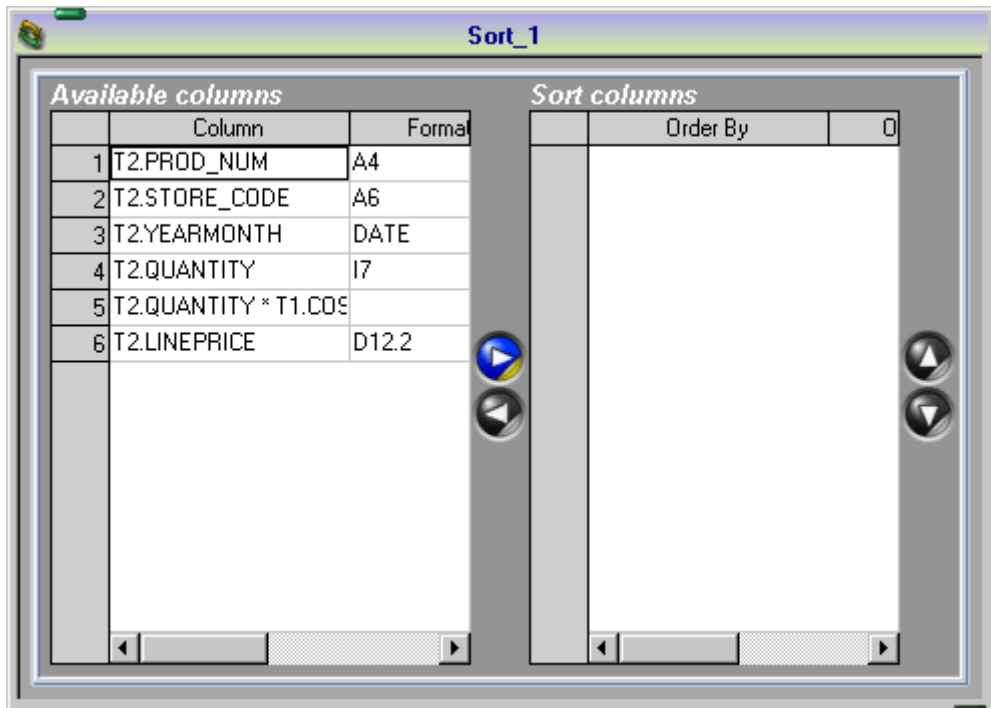
Select the column you want to sort by from the Available columns grid, and click the right arrow.

The column appears in the Sort columns grid.

4. Optionally, change the order in which the results will be sorted. To do this:
 - a. Click in the Order field.
 - b. Select an order from the drop-down list:

ASCENDING sorts the results from lowest to highest. This is the default.

DESCENDING sorts the results from highest to lowest.
5. Repeat steps 4 and 5 for each column you want to sort by.
6. Optionally, if you select more than one column, you can change the dominant sort by moving the columns up and down in the list using the up and down arrows.
7. Close the Sort window by double-clicking the control icon on the title bar.

Reference Sort Window

The Sort window contains the following fields/options:

Available Columns

Is a list of columns available in the data source.

Column

Is the column name.

Format

Is the column's format.

Alias

Is the alias name.

User defined description

Is a description of the column supplied by the user.

Sort Columns

Order By

Is the column that sorts the results.

Order

Specifies whether results are sorted in ascending or descending order.

Testing the SQL

Once you have created your query, you should test it to see if you are selecting the data you need. The `SQLAnswerSet` component enables you to retrieve a limited set of rows and review them as part of the process of designing your data mart.

You can also edit the SQL code. You can edit the SQL code with either SQL or FOCUS syntax.

Procedure How to Test the SQL

1. Add an `SQLAnswerSet` component to the workspace.
2. Connect the `SQLAnswerSet` component to the Column component, or to the Sort component if you are using one.
3. Double-click the `SQLAnswerSet` component to open it. The `SQLAnswerSet` window opens. For details on the `SQLAnswerSet` window see *SQLAnswerSet Window* on page 4-43.
4. Click *Test*.

After performing the query, ETL Manager places the retrieved data in the SQL Answer set grid or, if there is a problem with the SQL command, a message is displayed in the SQL Answer set grid.

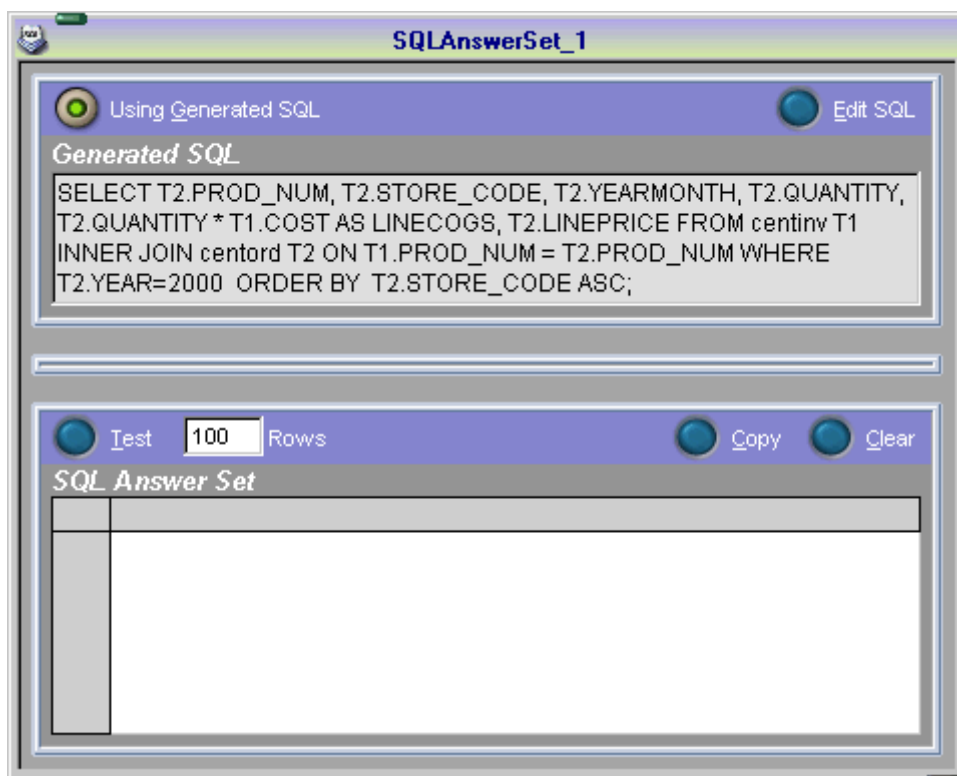
Procedure How to Edit SQL Code

1. Add an `SQLAnswerSet` component to the workspace.
2. Connect the `SQLAnswerSet` component to the Column component, or to the Filter or Sort component if you are using one.
3. Double-click the `SQLAnswerSet` component to open it. The `SQLAnswerSet` window opens. For details on the `SQLAnswerSet` window, see *SQLAnswerSet Window* on page 4-43.
4. Select *Edit SQL*.

The Custom SQL grid opens.

5. Make your changes to the SQL command.
 6. Click *Use Custom SQL* to specify that this is the code to be tested.
- Note:** The SQL command must conform to the SQL supported by the Server.

Reference **SQLAnswerSet Window**



The SQLAnswerSet window contains the following fields/options:

Using Generated SQL

Specifies the SQL generated by the request will be tested.

Edit SQL

Allows you to edit the SQL.

Generated SQL

Displays the current SQL.

Using Custom SQL

Specifies the SQL you edited will be tested.

Cancel

Cancels the changes made to the SQL.

Apply

Applies the changes made to the SQL.

Custom SQL

Displays the custom SQL.

Test

Performs the SQL test.

Rows

Specifies the number of rows to return.

Copy

Copies the contents of the results window to the clipboard.

Clear

Clears the results form the SQL Answer Set window.

SQL Answer set

Displays error messages or the answer set.

Specifying Transport Options

You can use FTP to move your data from the source subserver to the Server. You can specify this by selecting using the Transport component.

Note: This option is available for the Server on Windows NT and UNIX systems only.

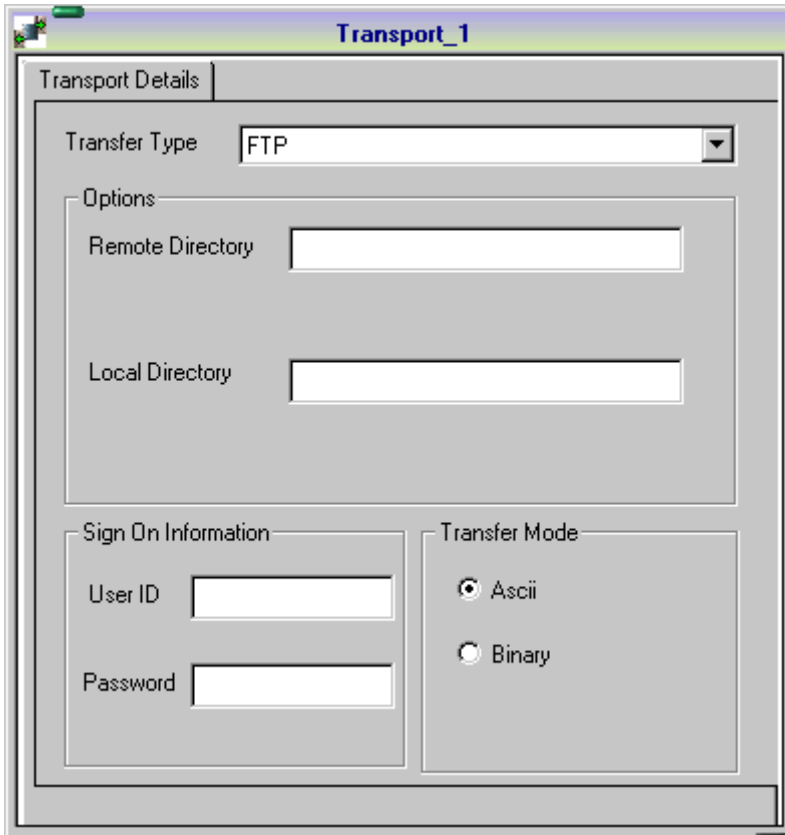
IWay does not provide FTP service and is only able to take advantage of the file transfer protocol if it already exists.

***Procedure* How to Set Transfer Details for FTP**

1. Add a Transport component to the workspace.
2. Connect the Transport component to the Column component, or to the Sort component if you are using one.
3. Double-click the Transport component to open it.
The Transport window opens. For details, see *Transport Window* on page 4-45.
4. Enter the directory where the file to transfer is located in the Remote Directory field.
5. Enter the directory where the file will be transferred to in the Local Directory field.

6. Enter a valid user ID in the User ID field.
7. Enter the password for the user ID in the Password field.
8. Select *ASCII* or *Binary* from the Transfer Mode section.

Reference Transport Window



The screenshot shows a window titled "Transport_1". Inside, there is a tab labeled "Transport Details". Below the tab, the "Transfer Type" is set to "FTP" in a dropdown menu. Under the "Options" section, there are two text input fields: "Remote Directory" and "Local Directory". The "Sign On Information" section contains two text input fields: "User ID" and "Password". The "Transfer Mode" section has two radio buttons: "Ascii" (which is selected) and "Binary".

The Transport window contains the following fields/options:

Transfer Type

Specifies that data will be transferred by FTP.

Options

Remote Directory

Is the directory of the file to transfer is located.

Local Directory

Is the directory to where the file will be transferred.

Sign On Information

User ID

Is a valid user ID.

Password

Is the password for the user ID.

Transfer Mode

Specifies whether data will be transferred in ASCII or Binary mode.

Selecting a Data Target

You need to specify the type and name of the data target to which your data will be copied. You can choose to use an existing table as your target, or create a new table.

- For details on creating a new target table, see *How to Create a New Data Target* on page 4-48.
- For details on using an existing table for your target, see *How to Select an Existing Data Target* on page 4-49.

A data target can be a relational data target, non-relational data target, a flat file, or a Server Transfer File. For details on these data target types see *Data Target Types* on page 4-47.

You can copy your data into as many as fifteen tables in a single request. When copying data into multiple targets, all targets must be of the same type. For example, data cannot be loaded into a flat file and relational data target in the same ETL Request. For details on loading data into multiple tables, see *How to Load Data Into Multiple Data Targets* on page 4-51.

After selecting a data target, you can do the following:

- Specify mappings. For details, see *Specifying Mappings* on page 4-56.
- Specify target options. For details, see *Specifying Data Target Options* on page 4-77.

Reference Data Target Types

The following data target types can be used:

- *Relational* data targets may include the following formats:
 - DB2
 - Informix
 - Ingres
 - Microsoft SQL Server
 - Nucleus
 - ODBC
 - ORACLE
 - Sybase
 - Teradata
- *Non-relational* data targets may include the following formats:
 - FOCUS
 - FUSION
- *Other* data targets may include the following:
 - Flat file
 - Server Transfer File. For details, see *Server Transfer Files* on page 4-47.

Reference Server Transfer Files

A Server Transfer File is a flat character file that is left at the remote server where the data source is located. In order for a Server Transfer File to work correctly, the following must be true:

- All data sources and Master Files must be available to the remote server where the transfer file will be written.
- Synonyms at the Server must have the same name and column names as the Master File at the source. This ensures that SQL designed by the ETL Workbench (while accessing the Hub Server) operates independently when sent directly to the source.
- The request must be able to use automatic pass through (APT).

You can ensure that Server Transfer Files work correctly by using the iWay Web Console to select data sources and build your Synonyms. Always choose the same name for your Synonyms. This is often less confusing for location transparency, and facilitates Server Transfer capability.

Procedure How to Create a New Data Target

1. Add a Target component to the workspace.

Connect the Target component to the Column component, or to the Sort or Transport component if you are using one.

2. Double-click the Target component to open it. The Target window opens. For details on the Target window see *Target Window* on page 4-53.

Note: Only data targets types that are configured on the Server will appear.

3. Click New *Target*.

The top of the Target window changes to reflect your choice.

4. Enter a name in the Table Name field in the following format

[database.] [owner.] name

where:

database

Is the name of the data target.

owner

Is the owner of the data target.

name

Is the name of the data target.

For relational data targets, if *database* or *owner* are not specified, the information is taken from the server profile.

For flat files, only the file name and format of the target operating system need to be specified.

5. Select a target type from the Select Target Type drop-down list.
6. For relational data targets, select the connection from the Connection drop-down list.
7. Optionally, enter a description in the Table description field.

8. Identify the key columns by clicking in the Keys field next to the target column you want to use as a key

Note: If you select the first column on the list, it will be the primary key. If you select a column that is lower on the list, the selected column and all columns above it become part of a concatenated primary key.

Procedure How to Select an Existing Data Target

1. Add a Target component to the workspace.

Connect the Target component to the Column component, or to the Sort or Transport component if you are using one.

2. Double-click the Target component to open it. The Target window opens. For details on the Target window, see *Target Window* on page 4-53.

Note: Only data targets types that are configured on the Server will appear.

3. Double-click a target to select it, or select the table and click *Select Table*.

The columns in the selected table display in the Target columns grid, and the Mapping and Transformation Rules grid opens below.

To select a multi-segment data target, see *How to Select a Multi-Segment Data Target* on page 4-49.

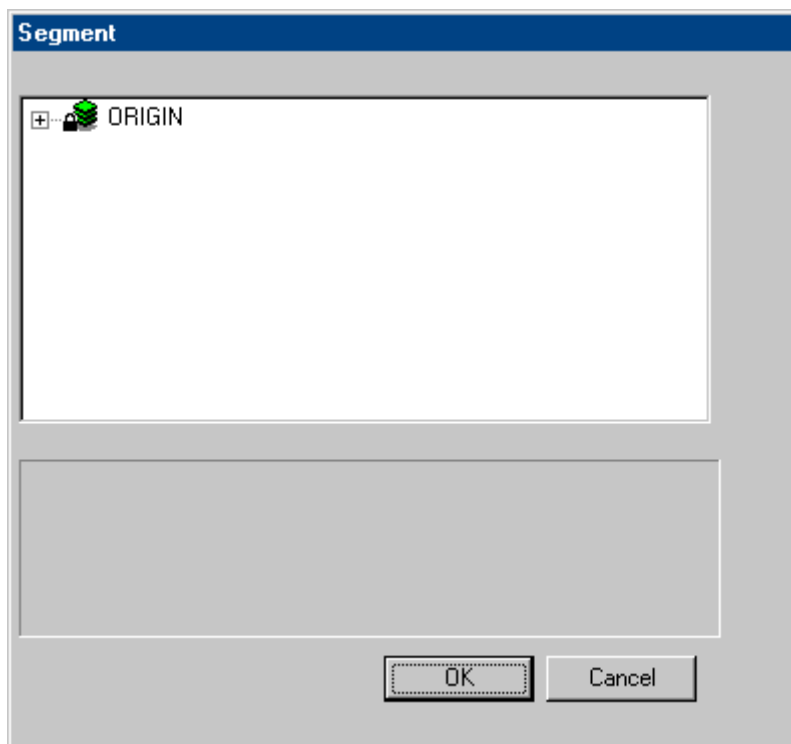
Procedure How to Select a Multi-Segment Data Target

1. Select a multi-segment FOCUS or Fusion data target.

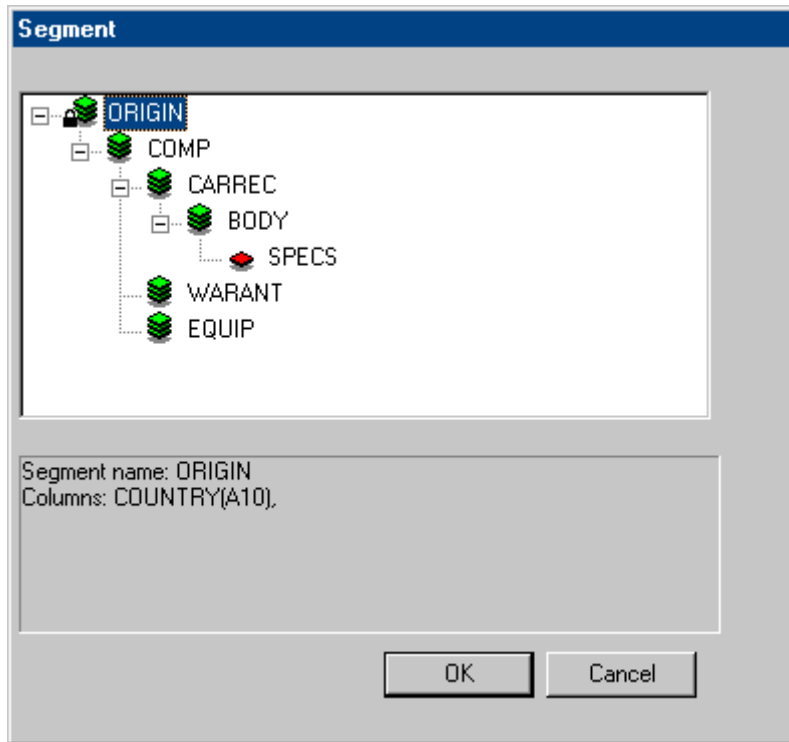
The button next to the Segment field turns green.

2. Click the *Segment* button.

The Segment dialog box opens displaying the top segment name:



- Click the plus sign to expand the list, and locate the segment you want to use.



- Select the segment you want to use, and click *OK*.

ETL Manager can load all segments along a single path. For example, in the CAR data source, if you select the SPECS segment, ETL Manager loads ORIGIN, COMP, CARREC, BODY, and SPECS.

Note: For single segment FOCUS or Fusion files, the segment is selected for you automatically.

Procedure How to Load Data Into Multiple Data Targets

- Add a Target component to the workspace.

Connect the Target component to the Column component, or to the Sort or Transport component if you are using one.

2. Double-click the Target component to open it. The Target window opens. For details on the Target window, see *Target Window* on page 4-53.

Note: Only data targets types that are configured on the Server will appear.

Select an existing data target, or create a new one. For details, see *How to Select an Existing Data Target* on page 4-49 and *How to Create a New Data Target* on page 4-48, respectively.

The columns in the selected table display in the Target columns grid, and the Mapping and Transformation Rules grid opens below.

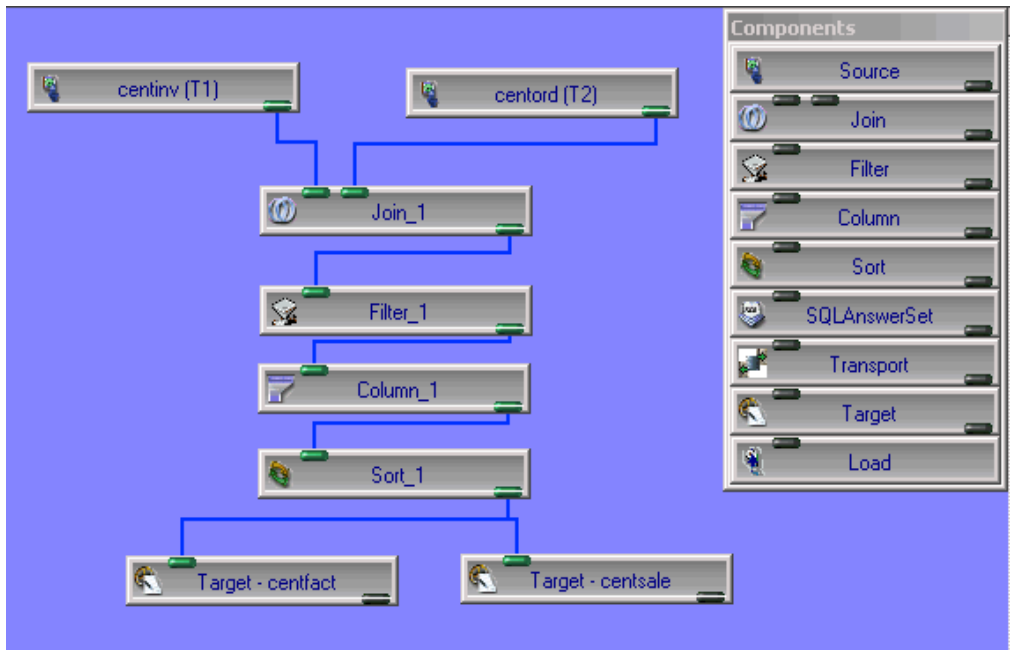
To select a multi-segment data target, see *How to Select a Multi-Segment Data Target* on page 4-49.

3. Add a second Target component to the workspace.

Connect the Target component to the Column component, or to the Sort or Transport component if you are using one.

4. Double-click the Target component to open it. The Target window opens.
5. Select an existing data target, or create a new one.

The workflow appears similar to the following:



Updating the List of Data Targets

You can refresh the list of available data targets and the list of columns in a data target. This is useful when you add data targets or columns to the iWay Catalog while building an ETL Request, and want these changes reflected in the data target or column list.

Procedure How to Update the List of Data Targets

In the Target window, click *Refresh Tables*.

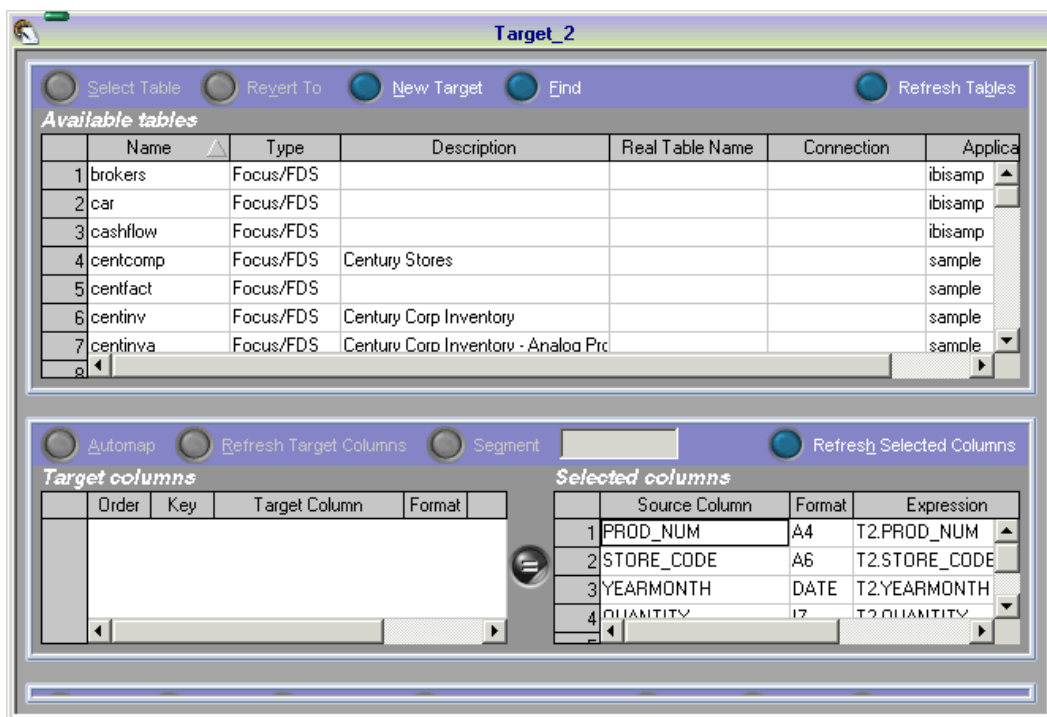
Procedure How to Update the List of Columns

In the Target window, click *Refresh Columns*.

Reference Target Window

The Target window contains the following fields/options:

Existing Table



Select Table

Selects the highlighted data target.

Revert to

Returns to the last selected data target.

New Target

Creates a new data target.

Find

Searches for a table by name, type, or description.

Refresh Tables

Updates the list of available data targets.

Available tables

Lists the available data targets.

Automap

Maps all columns with identical names and formats.

Refresh Target Columns

Updates the list of columns from the target descriptions.

Segment

For FOCUS and Fusion data targets, is the last segment to load.

Refresh Selected Columns

Updates the list of columns from the target descriptions.

Target columns

Lists the columns in the selected data target.

Selected columns

Lists the selected data target columns.

New Table

Existing Target

Table Name (Required) Target Type (Required) Connection

Table Description

Automap Refresh Target Columns Segment Refresh Selected Columns

Target columns					Selected columns				
	Key	Order	Target Column	Format		Source Column	Format	Expression	
1		1	PROD_NUM	A4	Prod	1	PROD_NUM	A4	T2.PROD_NUM
2		2	STORE_CODE	A6	Com	2	STORE_CODE	A6	T2.STORE_CODE
3		3	YEARMONTH	A8		3	YEARMONTH	A8	T2.YEARMONTH
4		4	QUANTITY	I7	Quar	4	QUANTITY	I7	T2.QUANTITY
5		5	LINECOGS	D16		5	LINECOGS	D16	T2.QUANTITY * T1.0
6		6	LINEPRICE	D12.2	Line	6	LINEPRICE	D12.2	T2.LINEPRICE

Existing Target

Allows you to use an existing table for the data target.

Table Name (Required)

Is the data target name.

Target Type (Required)

Is the data type of the new data target.

Connection

Is the connection for the new data target. For a relational data target, this is a server. For ODBC, this is a data source.

Table Description

Is a description for the data target.

Automap

Maps all columns with identical names and compatible formats.

Refresh Target Columns

Refreshes the lists of the target columns.

Refresh Selected Columns

Refreshes the lists of the selected columns.

Target columns

Lists the columns in the selected data target.

Selected columns

Lists the selected data target columns.

Specifying Mappings

After selecting a data source and target, you can create mappings to transform your data in the following ways:

- Map the columns of the source, or combinations of columns in the source, to columns in the data target. This is for non-flat file and non-Server Transfer File target formats.
- Convert the extract data into the form necessary for the data target.
- Create new columns for flat file and Server Transfer File targets.

There are three methods for mapping columns:

- **Automapping** automatically maps together columns with the same name.
- **Simple mapping** maps one column in the source to one in the target.
- **Complex mapping** builds expressions.


Procedure How to Specify Automapping

In the Target window, click Automap.

All columns with identical names and compatible formats in the Target columns grid and Selected columns grid display as equal to each other in the Mapping and Transformation Rules grid.

For all columns which do not have a match, you must use simple or complex mappings. For details, see *How to Specify a Simple Mapping* on page 4-57 and *How to Specify a Complex Mapping* on page 4-57.

Procedure How to Specify a Simple Mapping

In the Target window, select a column from the Target columns grid, select a column from the Selected columns grid, and click .

The source and target columns display as equal to each other in the Mapping and Transformation Rules grid.

Procedure How to Specify a Complex Mapping

1. Double-click a column in the Target columns grid.
The column appears in the Mapping and Transformation Rules grid.
2. Select the mapping in the Mapping and Transformation Rules grid, and click Calculator.
The Transform calculator opens.
3. Enter an expression with the calculator, and click OK. For details on using the Transform calculator, see *Using the Transform Calculator* on page 4-60.

The new mapping appears in the Mapping and Transformation Rules grid.

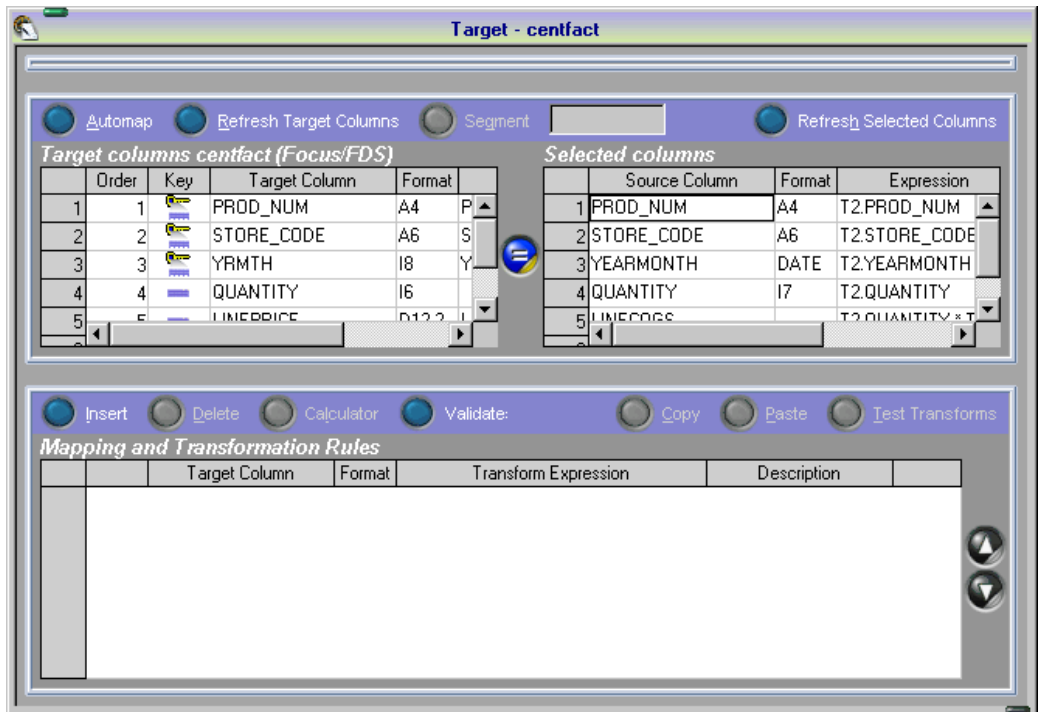
Procedure How to Specify a Mapping in the Mapping and Transformation Rules Grid

1. Double-click a column in the Target columns grid.
The column appears in the Mapping and Transformation Rules grid.
2. Select the mapping in the Mapping and Transformation Rules grid, and double-click in the Transform Expression field.
3. Enter the expression you want to use for the mapping.

Procedure How to Test Mappings

1. In the Mapping window, click *Test*.
The Test Transforms Results window opens, displaying the results of your mappings.
2. Click *OK* to close the window, or *Copy* to copy the answer set to the clipboard.

Reference Target Window (Mapping)



The Target window (mapping) contains the following fields/options:

Automap

Maps all columns with identical names.

Refresh Target Columns

Updates the list of available target columns.

Refresh Selected Columns

Updates the list of selected columns.

Target columns

Are the columns in the data target.

For flat files and Server Transfer Files, these are the same columns that are listed in the Selected columns grid.

Note: You can distinguish between columns you have created and columns that already exist in the target with the icons that display with each column.



Is a key column.



Is a column.



Is a virtual column described in an extract transform.



Is a virtual column described in the file description.

Selected columns

Are the columns of data in the data source.

Insert

Inserts a temporary column.

Delete

Deletes a selected mapping.

Calculator

Opens the Transform calculator.

Validate

Applies a transformation that screens records as they are loaded into the data target. If a record does not meet the criteria, it is rejected.

Copy

Copies selected transforms.

Paste

Pastes copied transforms to the Mapping and Transformation Rules grid.

Test transforms


Retrieves up to 100 records and 4096 characters per row from the source and applies all available transforms. This option enables you to review the data being copied and ensure that the syntax of each transformation is correct. No duplicate processing is performed. For example, if a key value is duplicated, a unique constraint violation may occur. Similarly, if a value is missing for a column described as not null, a constraint violation will occur at run time.

If there are any errors, ETL Manager displays an error message and marks the incorrect transformation with a red X.

Mapping and Transformation Rules

Are the columns that you have mapped together (including any transformations you have applied), and the new columns you have created.

Note: You can distinguish between columns you have created and columns that already exist in the target with the icons that display with each column.

 Is a column.

 Is temporary column.

Creating a Temporary Column for a Mapping

You can create a temporary column within the mapping step to add intermediate calculations that are not part of the original SQL command or target data mart. This is often useful for complex calculations that are actually composed of multiple expressions. Creating a temporary column makes the transformation easier to understand and support. It is also helpful in performing data type conversions when multiple steps are required.

Procedure How to Create a Temporary Column

1. In the Target window, click *Insert*.

A new column appears in the Mapping and Transformation Rules grid.

2. Double-click in the Target Column field and enter a name for the temporary column.
3. Double-click the Format field and enter a format for the temporary column.

You can create a complex expression for your temporary column. For more information, see *Using the Transform Calculator* on page 4-60.

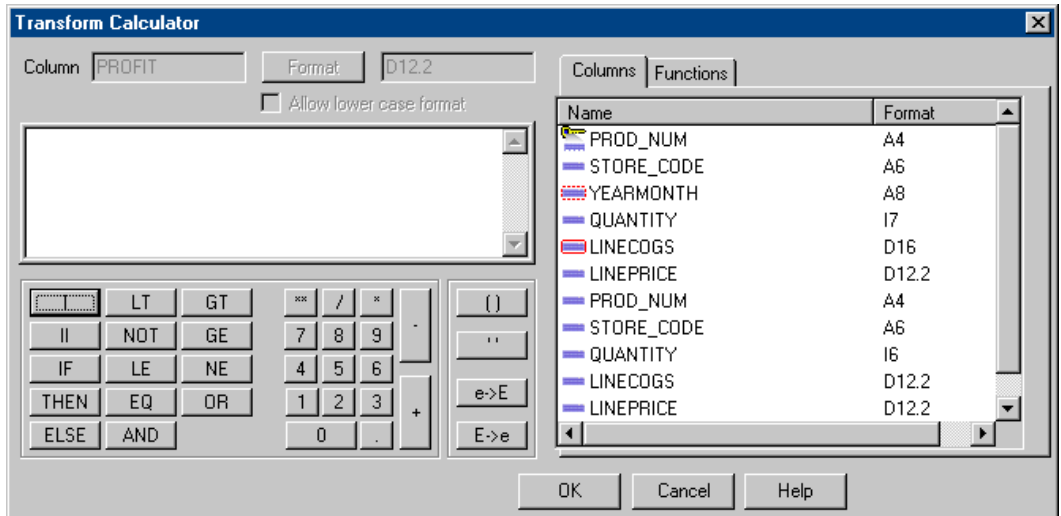
Note: You can tell your temporary columns apart by the icons that appear next to them.

Tip: Since calculations are performed in the order they appear, make sure that your temporary column appears *before* any column in the Transform Rules box that uses it to calculate a value.

Using the Transform Calculator

You can use the Transform calculator to create complex mappings for your ETL Request. The Transform calculator is accessed by clicking Calculator on the Target (mapping) window.

Reference Transform Calculator



The Transform calculator contains the following fields/options:

|

Concatenates two values, retaining any trailing blanks after the first one. For example, if FIRST_NAME and LAST_NAME were both in A15 format, the expression

```
FULL_NAME = FIRST_NAME | LAST_NAME
```

would produce a column like the following:

```
MICHAEL      SMITHSON
TERRI        WHITE
```

```
.
.
.
```

||

Concatenates two values, suppressing any trailing blanks in the first. For example, to construct the full name and insert a comma, the syntax

```
FULL_NAME = LAST_NAME || ( ' , ' | FIRST_NAME )
```

would produce a column like the following:

```
SMITHSON, MICHAEL  
WHITE, TERRI
```

·
·
·

The concatenation in the parentheses is done first (preserving the blank space after the comma), and the result is then concatenated to LAST_NAME, suppressing the trailing blanks of LAST_NAME.

IF

Establishes a conditional test.

THEN

Specifies the action to perform if the result of a conditional test is TRUE.

ELSE

Specifies the action to perform if the result of a conditional test is FALSE.

LT

Returns the value TRUE if the value on the left is less than the value on the right.

LE

Returns the value TRUE if the value on the left is less than or equal to the value on the right.

EQ

Returns the value TRUE if the value on the left is equal to the value on the right.

AND

Returns the value TRUE if both operands are true.

NOT

Returns the value TRUE if the operand is false.

GT

Returns the value TRUE if the value on the left is greater than the value on the right.

GE

Returns the value TRUE if the value on the left is greater than or equal to the value on the right.

NE

Returns the value TRUE if the value on the left is not equal to the value on the right.

OR

Returns the value TRUE if either operand is true.

Raises a value to the specified power.

/

Divides values.

If you attempt to divide by 0, ETL Manager sets the value of the expression to 0.

Multiplies values.

-

Subtracts values.

+

Adds values.

0-9

Inserts numbers into the expression box.

()

Inserts parentheses into the expression box. You use parentheses to determine the order in which ETL Manager performs calculations (see *Order of Evaluation* on page 4-64).

"

Inserts two single quotation marks into the expression box (used for enclosing test values).

e->E

Converts selected text in the expressions window to uppercase.

E->e

Converts selected text in the expressions window to lowercase.

Columns

Lists the source columns from your query. Click the name of any source column to insert it into the expression box. For more information on formats, see the FOCUS user's manual for your server platform.

Functions

Displays the available functions. For more information, see *Using Functions* on page 4-65.

Note: You can perform some of these mappings (for example, numeric to alphanumeric) using the EDIT function in the Transform calculator.

Order of Evaluation

The order of evaluation can affect the result of the expression. ETL Manager evaluates numeric and Boolean operators from left to right in the following order:

- Operations enclosed in parentheses
- Exponentiation
- Multiplication and division
- Addition and subtraction
- EQ, NE, LE, LT, GE, GT, NOT
- AND
- OR

Example Changing the Order of Evaluation

The expression

```
RETAIL_PRICE - UNIT_COST * UNIT_SOLD
```

gives an incorrect result because UNIT_SOLD is first multiplied by UNIT_COST, and then the result is subtracted from RETAIL_PRICE.

You can change the order of evaluation and improve readability by enclosing expressions in parentheses. An expression in parentheses is evaluated first. The correct expression for the preceding example is:

```
(RETAIL_PRICE - UNIT_COST) * UNIT_SOLD
```

Using Functions

When you click the *Functions* tab in the Transforms calculator, a list of available functions appears.

Note: All functions may not be available on all platforms.

For more information on these functions, see the ETL Manager online help and the *Using Functions* manual.

Function	Arguments	Description
ABS	<i>(field)</i>	Returns the absolute value of its argument.
ARGLEN	<i>(len, data, output)</i>	Measures the length of a character string within a field, excluding trailing blanks.
ATODBL	<i>(value, len, return)</i>	Converts a number in alphanumeric format to decimal format.
AYM	<i>(oldate, ins, output)</i>	Adds or subtracts months from dates stored in alphanumeric or integer format.
AYMD	<i>(oldate, ins, output)</i>	Adds or subtracts a given number of days to a date in [YY]YYMMDD format.
BITSON	<i>(bitnumber, string, outfield)</i>	Evaluates an individual bit within a character string to determine whether it is on or off.
BYTVAL	<i>(string, startbit, number, outfield)</i>	Evaluates a string of bits within character strings and returns its binary value.
BYTVAL	<i>(char, outfld)</i>	Translates a character to its corresponding ASCII code.
CHGDAT	<i>(infmt, outfmt, input, output)</i>	Rearranges the year, month, and day components of a date and converts a date between long and short date formats.
CHKFMT	<i>(numchar, infield, mask, outfield)</i>	Checks numeric and alphanumeric fields for invalid character types.
CTRAN	<i>(inlen, infld, decfrm, decto, output)</i>	Replaces characters in a string, taking decimal ASCII values as arguments for the target.
CTRFLD	<i>(infld, inlen, outfld)</i>	Centers a character string within a field.

Function	Arguments	Description
DADMY	<i>(date, idays)</i>	Returns date in day-month-year format given the number of days since 1/1/1900.
DADYM	<i>(date, idays)</i>	Given the number of days since 1/1/1900, returns date in day-year-month format.
DAMDY	<i>(date, idays)</i>	Returns date in month-day-year format given the number of days since 1/1/1900.
DAMYD	<i>(date, idays)</i>	Returns date in month-year-day format given the number of days since 1/1/1900.
DATEADD	<i>(date, 'unit', #units)</i>	Adds or subtracts years, months, or days to or from a date.
DATECVT	<i>(date, 'infmt', 'outfmt')</i>	Converts dates from one date format to another.
DATEDIF	<i>(from_date, to_date, 'unit')</i>	Calculates the difference between two dates, expressed as years, months, or days.
DATEMOV	<i>(date, 'move-point')</i>	Moves a date to a significant point on the calendar.
DAYDM	<i>(date, idays)</i>	Returns date in year-day-month format given the number of days since 1/1/1900.
DAYMD	<i>(date, idays)</i>	Returns date in year-month-day format given the number of days since 1/1/1900.
DECODE	<i>fieldname(code1 result1... [ELSE default])</i>	Changes a variable to an associated name.
DMOD	<i>(dividend, divisor, outfld)</i>	Returns the remainder from a division as a number in decimal format.
DMY	<i>(begin, end)</i>	Returns the difference between two dates in day-month-year order.
DOWK	<i>(date, output)</i>	Returns the day of the week in A4 format.
DOWKL	<i>(date, output)</i>	Returns the day of the week in A12 format.

Function	Arguments	Description
DTDMY	<i>(days, output)</i>	Given the date in day-month-year format, calculates the number of days since 1/1/1900.
DTDYM	<i>(days, output)</i>	Calculates the number of days since 1/1/1900 given the date in day-year-month format.
DTMDY	<i>(days, output)</i>	Calculates the number of days since 1/1/1900 given the date in month-day-year format.
DTMYD	<i>(days, output)</i>	Calculates the number of days since 1/1/1900 given the date in month-year-day format.
DTYDM	<i>(days, output)</i>	Calculates the number of days since 1/1/1900 given the date in year-day-month format.
DTYMD	<i>(days, output)</i>	Calculates the number of days since 1/1/1900 given the date in year-month-day format.
EDIT	<i>(field, ['mask'])</i>	Converts between alphanumeric and integer data. Used with a mask, extracts characters from an alphanumeric string and inserts new characters.
EXP	<i>(power, outfield)</i>	Raises the number “e” to a power you specify.
EXPN	<i>(n.nn {E D} {+ -} p)</i>	Evaluates an argument expressed in scientific notation.
FGETENV	<i>(varlength, varname, outfieldlen, outfldformat)</i>	Gets the value of an environmental variable and returns it as an alphanumeric string.
FIND	<i>field = FIND(fieldname [AS dbfield] IN file;</i>	Verifies if a value exists in an indexed field in another file.
FMOD	<i>(dividend, divisor, outfld)</i>	Returns the remainder from a division as a number in floating-point format.

Function	Arguments	Description
FPUTENV	<code>(namelength, 'name', valuelength, value, outfield)</code>	Assigns a character string to an environmental variable.
FTOA	<code>(input, '(format)', output)</code>	Converts numbers from decimal format to alphanumeric format.
GETTOK	<code>(input, inlen, tokeno, delim, outlen, output)</code>	Divides a string into tokens and returns the specified token.
GETUSER	<code>(outfield)</code>	Retrieves the user ID from the system.
GREGDT	<code>(datej, dateg)</code>	Converts a Julian date to a Gregorian date.
HEXBYT	<code>(input, output)</code>	Converts a numeric value to its corresponding ASCII character.
HADD	<code>(dtfield, 'component', increment, length, 'format')</code>	Increments a date-time field by a given number of units.
HCNVRT	<code>(value, '(fmt)', length, 'outputfmt')</code>	Converts a date-time field to alphanumeric format for use with operators such as EDIT, CONTAINS, and LIKE.
HDATE	<code>(dtfield, 'format')</code>	Extracts the date portion of a date-time field and converts it to a date format.
HDIFF	<code>(date1, date2, 'component', 'format')</code>	Finds the number of boundaries of a given type crossed going from date 2 to date 1.
HDTTM	<code>(date, length, format)</code>	Converts a date field to a date-time field. The time portion is set to midnight.
HGETC	<code>(length, 'format')</code>	Stores the current date and time in a date-time field.
HHMMSS	<code>(output)</code>	Returns the current time in alphanumeric format.
HINPUT	<code>(inputlength, 'inputstring', length, 'Hfmt')</code>	Converts an alphanumeric string to a date-time value.
HMIDNT	<code>(value, length, 'format')</code>	Changes the time portion of a date-time field to midnight (all zeroes).

Function	Arguments	Description
HNAME	<i>(value, component, format)</i>	Extracts a specified component from a date-time field and returns it in alphanumeric format.
HPART	<i>(value, 'component', 'format')</i>	Extracts a specified component from a date-time field and returns it in numeric format.
HSETPT	<i>(dtfield, 'component', value, length, 'Hformat')</i>	Inserts the numeric value of a specified component into a date-time field.
HTIME	<i>(length, value, 'format')</i>	Converts the time portion of a date-time field to a numeric number of milliseconds (if the first argument is 8) or microseconds (if the first argument is 10).
IMOD	<i>(dividend, divisor, outfld)</i>	Returns the remainder from a division as a number in integer format.
INT	<i>(field)</i>	Returns the integer part of its argument.
ITONUM	<i>(maxbytes, infield, outfield)</i>	Converts large binary integers in non-FOCUS files to double-precision format.
ITOPACK	<i>(maxbytes, infield, outfield)</i>	Converts large binary integers in non-FOCUS files to packed-decimal format.
ITOZ	<i>(outlength, number, outfield)</i>	Converts numbers from numeric format to zoned format for extract files.
JULDAT	<i>(dateg, datej)</i>	Converts a Gregorian date to a Julian date.
LAST	<i>fieldname</i>	Retrieves the previous value for a field.
LCWORD	<i>(inlength, infield, outfield)</i>	Converts the letters in a given string to mixed case: the first letter of each new word and the first letter after a single or double quotation mark are converted to uppercase; the rest to lowercase.
LJUST	<i>(length, input, output)</i>	Left-justifies a character string within its field.

Function	Arguments	Description
LOCASE	<i>(inlength, infield, outfield)</i>	Returns the infield in lowercase.
LOG	<i>(field)</i>	Returns the natural logarithm of its argument.
MAX	<i>(field, field)</i>	Returns the maximum value from its list of arguments.
MDY	<i>(begin, end)</i>	Returns the difference between two dates in month-day- year order.
MIN	<i>(field, field)</i>	Returns the minimum value from its list of arguments.
NORMSDST	<i>(value, 'D8')</i>	Calculates the percentage of data values that are less than or equal to a normalized value.
NORMSINV	<i>(value, 'D8')</i>	Finds the normalized value that forms the upper boundary of a percentile in a standard normal distribution curve.
OVRLAY	<i>(field, len, string, len, pos, output)</i>	Overlays a substring to a specified place on another character string.
PARAG	<i>(inlen, infld, delim, parlen, output)</i>	Divides lines of text into smaller lines by marking them off with a delimiter character.
PCKOUT	<i>(infield, outlength, outfield)</i>	Writes packed numbers of varying lengths (between one and 16 bytes) to extract files.
POSIT	<i>(field, len, string, len, output)</i>	Finds the starting position of a substring within a larger string.
RANDNUM	<i>(outfield)</i>	Returns a random number between 0 and 32768.
REVERSE	<i>(length, input, output)</i>	Reverses the input characters.
RJUST	<i>(length, input, output)</i>	Right-justifies a character string within a field.
ROUND	<i>(field)</i>	Rounds a floating-point number.

Function	Arguments	Description
<code>SIMILAR</code>	<code>(inlen1, str1, inlen2, str2, outfld)</code>	Matches two character strings for a degree of likeness.
<code>SOUNDEX</code>	<code>(inlength, infield, outfield)</code>	Converts character strings to 4-letter codes, enabling you to search for character strings phonetically without knowing how they are spelled.
<code>SPACE</code>	<code>(drive, freespace)</code>	Determines whether a particular disk drive is ready, and the number of bytes of disk space available.
<code>SPELLNM</code>	<code>(inlength, infield, outfield)</code>	Takes a number with two decimal spaces and spells it out with dollars and cents.
<code>SQRT</code>	<code>(field)</code>	Returns the square root of its argument.
<code>SQUEEZ</code>	<code>(length, string, outfield)</code>	Reduces multiple contiguous blank characters within an input string to a single blank character.
<code>STRIP</code>	<code>(length, string, char, outfield)</code>	Removes all occurrences of a specific character from an input string.
<code>SUBSTR</code>	<code>(inlen, infld, start, end, outlen, outfld)</code>	Extracts a substring of a given length and starting position from a larger string.
<code>SYSTEM</code>	<code>(length, 'string ', returncode)</code>	Calls a DOS program, a DOS batch program, or a Windows NT/2000 application.
<code>TIMETOTS</code>	<code>(time, length, format)</code>	Converts its time argument to a timestamp, using the current date to supply the date component of its value.
<code>TODAY</code>	<code>(output)</code>	Returns the current date in alphanumeric format (MM/DD/YYYY).
<code>TRIM</code>	<code>(trim_where, string, string_length, pattern, pattern_length, outfield)</code>	Removes leading and/or trailing occurrences of a pattern within a string.
<code>TRIMLEN</code>	<code>(string)</code>	Returns the length of the argument string excluding trailing blanks.

Function	Arguments	Description
UPCASE	<i>(len, string, output)</i>	Converts all characters in string to uppercase.
UFMT	<i>(string, inlength, outfield)</i>	Converts characters in alphanumeric field values to hexadecimal (HEX) representation.
YM	<i>(fromdate, todate, diff)</i>	Returns the number of days between two dates.
YMD	<i>(begin, end)</i>	Returns the difference between two dates in year-month-day order.

Field Formats and Mapping

There are three types of field formats:

- **Alphanumeric format** is values composed of letters, numbers, or special characters.
- **Date and time format** is dates and times or date and time components.
- **Numeric format** is values composed of the digits 0 through 9 and, optionally, a minus sign. There are four numeric formats:
 - **Decimal** is used for whole numbers or fractions, using 8 bytes of storage and a maximum number of 15 positions. This type of field automatically inserts a comma after every third significant digit.
 - **Floating Point** is whole numbers or fractions, using 4 bytes of storage and a maximum number of 9 positions.
 - **Integer** is whole numbers, using 4 bytes of storage and a maximum number of 9 positions. Numbers after the decimal are truncated.
 - **Packed** is whole numbers or fractions, using 8 bytes of storage and a maximum number of 15 positions. This type of field will automatically round any number with more than the specified number of decimal places to the number of decimal places in the format. This is recommended for monetary values.

You can format a field as an alphanumeric, date, or numeric field. For details, see *Formatting New Fields* on page 4-73.

Reference Rules for Mapping and Field Formats

The columns you map together must have the same format, or ETL Manager must be able to translate the source query format into the target format automatically. Details are shown in the following table:

If the Target format is...	...and the Source query format is...	Can you map?
A11	A10	Yes
A10	A11	No. Use the EDIT or SUBSTR functions to make this work.
D6.2	I5	Yes
I5	D6.2	Yes, but you will lose the numbers after the decimal.
A10	D6.2	No
D6.2	A10	Yes, provided that only digits 0-9 and a period are provided in the character input.

Formatting New Fields

If you are creating a new or virtual field, you assign the field a format that you choose. You can assign a format in two ways:

- In the Format field in the Mapping and Transformation Rules grid by typing the format.
- With the Format dialog box. This is accessed by clicking the Format button in the Transform calculator.

Procedure How to Format a Field as Alphanumeric

1. In the Format dialog box, choose *Alphanumeric* from the Types list.
2. In the Specifications box, enter a length for your alphanumeric field (from 1 to 256 characters).

Procedure How to Format a Field as Date and Time

1. In the Format dialog box, choose *Date* from the Types list.
2. In the Specifications box, choose a date format from the drop-down list. You have the following choices (the example illustrates how you would enter 9:30 AM on October 15, 2002):

Format	Description	Example
D	Day.	15
DMY	Day/Month/2-digit year.	15/10/02
DMYY	Day/Month/4-digit Year.	15/10/2002
JUL	Date in Julian format. A 5-digit number where the first two numbers are the 2-digit year and the last three numbers are the number of the day counting from January 1.	98288
M	Month.	10
MDY	Month/Day/2-digit Year.	10/15/02
MDYY	Month/Day/4-digit Year.	10/15/2002
MY	Month/2-digit Year.	10/2002
MY Y	Month/4-digit Year.	10/2002
Q	Quarter.	4
QY	Quarter 2-digit Year.	Q4 02
QYY	Quarter 4-digit Year.	Q4 2002
W	Day of week where Monday=1.	6
Y	2-digit year.	02
YM	2-digit Year/Month.	02/10
YMD	2-digit Year/Month/Day.	02/10/15
YY	4-digit year.	2002
YYM	4-digit Year/Month.	2002/10
YYMD	4-digit Year/Month/Day.	2002/10/15

Format	Description	Example
HYYMDI	4-digit year/Month/Date Hours:Minutes.	2002/10/15 9:30
HYYMDS	4-digit year/Month/Date Hours:Minutes:Seconds.	2002/10/15 9:30:10
HYYMDs	4-digit year/Month/Date Hours:Minutes:Seconds:Milliseconds. YYYYMMDDHHMMSSsss	2002/10/15 9:30:10:30
HYYMDm	4-digit year/Month/Date Hours:Minutes:Seconds:Nanoseconds. YYYYMMDDHHMMSSssssss	2002/10/15 9:30:10:30
HHIS	HHMMSS	9:30:10

Note: Time formats (data type H) apply only to FOCUS reports. Reports executed in the ETL Manager test window cannot use these formats.

Procedure How to Format a Field as Numeric

1. In the Format dialog box, choose *Decimal*, *Floating Point*, *Integer*, or *Packed* from the Types list.
2. In the Specifications box, enter a width in the Width field. The width is the maximum number of positions in your field including all the digits, the decimal point, any edit options and, if necessary, a minus sign.
3. Enter the number of places allowed after the decimal point for your number in the Precision field. The maximum number that you can enter depends on what format you chose.

Notice that since, by definition, integers do not have a decimal point, the Precision field is eliminated for this type.

4. In the Edit Options box, select any combination of the following:
 - B -- Bracket negative* encloses a negative number in parentheses.
 - C -- Comma inclusion* inserts a comma after every third significant digit. This option takes effect automatically for Decimal format.
 - E -- Scientific notation* displays the number in scientific notation.
 - L -- Leading zeros* adds leading zeros to the full field length.
 - M -- Floating dollar* adds a dollar sign immediately to the left of the number and inserts a comma after every third significant digit.
 - R -- Credit negative* adds the characters CR after a negative number.
 - S -- Zero suppression* suppresses leading zeros; displays a space if the value is 0.

Converting Legacy Character Date Fields Into Date Format

This topic describes how to convert legacy character date fields into numeric offset date format. (An example of a date format is December 1, 1900.)

Suppose the source data contains the column ORD_DATE, which is stored in ten character positions (for example, 1996/11/22). You want to map this data to a column in the data target named ORDER_DATE. This column in the relational data warehouse has a DATE format.

This technique uses the following steps.

1. Removes the slashes from this *old*-style date field.
2. Converts the data into an integer representation, with date formatting.
3. Converts the data into the common iWay definition for offset dates.
4. Moves the data into the final target column:

```
COBEDIT/A8          = EDIT(ORD_DATE, '9999$99$99')
INT_COBOL/I8YYMD    = EDIT(COBEDIT)
ORDER_DATE          = INT_COBOL
```

Note: Each line above is a separate transform.

Validating Records

You can create a business rule that screens records as they are loaded into the data target. If a record does not meet the criteria, it is rejected.

When a value meets the criteria of the rule, a value of 1 is assigned to the record and the record is loaded. If the value does not meet the criteria of the record, the value of 0 is assigned to the record and the record is rejected.

Procedure How to Validate Records

1. In the Target (Mapping) window, click Validate.

A column is added containing <VALIDATE> in the Column name field. An *f* is added to the first field of the next row. All expressions in this row and the rows beneath it will be rules that are applied to records as they are loaded.

2. Enter the column you want to apply the rule to in the Column Name field.
3. Enter the format in the Format field.
4. Enter the rule to apply to the specified in the Expressions box, or click Calculator to create an expression with the calculator.
5. Optionally, enter a description in the Description field.

Specifying Data Target Options

After selecting a target table, you can select the options that govern the behavior of the ETL Request when copying data into the data target. These options can be specified in the Load component, and include the following:

- Specifying data target options. For details, see *Specifying Target Setup Options* on page 4-77.
- Setting up output parameters. For details, see *Setting Up Output Parameters* on page 4-83.
- If you are using the bulk load procedure, specifying options for the bulk load. For details, on bulk loading see *Using Bulk Loading* on page 4-89.
- Optionally, specifying options for logging records. For details, see *Logging Records* on page 4-102.

Specifying Target Setup Options

You can specify the target setup options which govern the behavior of ETL processing as data is copied to the data target.

Procedure How to Specify Target Setup Options

1. Add a Load component to the workspace, and connect it to the Target component.
2. Double-click the Load component to open it.

The Load window opens with the Target Setup tab open. For details, see *Specifying Target Setup Options* on page 4-77.

3. If your table is not a flat file or Server Transfer file, select a behavior for the key matching logic when duplicate rows exist from the Key Matching Logic If duplicate row exist section. The options are:

Reject row rejects the row if the row is found.

Include row does not check for duplicates and inserts the rows dynamically.

Update row deletes the row if the key value on the input row is found in the table. If you select *Update Row*, Key Matching Logic if Row does not exist appears. Select an option.

Delete row deletes the row if the key on the input row (or entire row if no keys) is found in the table.

4. Select the processing options you want from the Option section. The options are:

Stop processing when 0 records detected considers an ETL Request that does not retrieve any rows to have failed, and no additional dependencies are run.

Use alternate load procedure specifies bulk load is to be used. This is available only for selected relational data targets. If you select this options, you must set additional options. For details, see *Using Bulk Loading* on page 4-89.

Delete all rows prior to load deletes all rows prior to loading data.

Commit transactions every ____rows specifies how often to commit transactions.

5. Select whether to remove data prior to loading the data target in the Prior to Load section. The options are:

Leave Table does not delete the rows already in a data target.

Drop Table drops and recreates the data target. This option is available only for FOCUS and Fusion data targets.

Delete rows from table deletes all rows prior to loading data. A DELETE FROM TABLE command is issued against the target, and a log of the deleted records is created so that a Rollback command can later be issued to recover them. This value is not available for FOCUS and Fusion data targets.

Truncate logically empties the table. A log is not created, so this is often faster than selecting *Delete rows from table*. However it requires privileges that may not be in effect, so if the Truncate command fails, the result is as if you had chosen to delete the rows. This value is not available for FOCUS and Fusion data targets.

Reference Target Setup Tab

The screenshot shows a window titled "Load_1" with four tabs: "Target Setup", "Output Parameters", "Bulk Load", and "Record Logging". The "Target Setup" tab is active. It contains the following options:

- ☐ Enable Change Data Capt
- Key Matching Logic**
If duplicate rows exist
 - ☒ Reject Row
 - ☐ Include Row
 - ☐ Update Row
 - ☐ Delete Row
- Options**
 - ☒ Stop processing if 0 rows detected
 - ☐ Use alternate load procedure
 - Commit transactions on every rows
- Prior to load**
 - ☒ Leave Table
 - ☐ Delete rows from table
 - ☐ Truncate

The Target Setup tab contains the following fields/options:

Enable Change Data Capture

Enables the use of change data capture, an optional component of ETL Manager.

Key Matching Logic If duplicate rows exist

Specifies ETL Manager's behavior while loading data if duplicate rows exist. This is not available if you are using an alternate load protocol (specified by selecting the Use alternate transfer protocol checkbox).

Reject row

A SELECT command is issued against the target table to see if a row exists with the key values. If there are no key columns, ETL Manager screens all columns. If the row is found, the row is rejected.

Include row

Includes the duplicate row in the data target.

Update row

If the key value on the input row, or entire row if there are no keys, are found in the table, ETL Manager updates these rows. All non-key values are updated in the target. If you select this option, Key Matching Logic If row does not exist appears in the tab. Select to Include Row or Reject Row.

Delete row

If the key on the input row, or the entire row if there are no keys, is found in the table, that row is deleted.

Note: The Reject, Update, and Delete options can adversely affect performance because they determine the existence of a key value on an incoming row *before* performing the specified action. This is done by issuing a SELECT command against the target table, then waiting for the response. If one of these actions is required, try to limit the volume of rows in the incremental change. These actions will perform best if there are unique keys on the table.

If row does not exist

Determines whether to include keys that do not exist in the data target. This option is available when the **Update row** or **Delete row** options are selected in the Key Matching Logic section.

Include row

Includes the key in the data target.

Reject row

Does not include the key in the data target.

Options

Determines the processing method used for the ETL Request.

Stop processing when 0 records detected

A ETL Request that does not retrieve any rows is considered to have failed, and no additional remote procedures and dependencies are run. Do not select this if you normally run requests that do not retrieve rows.

Use alternate load procedure

Specifies bulk load is to be used. This is available only for relational data targets.

Delete all rows prior to load

Deletes all rows prior to loading data. A DELETE FROM TABLE command is issued against the target. While this is an SQL standard for most relational systems, it may not always be the best performer. Some RDBMS support a “truncate” command that logically empties the table. Truncate, however, often requires special privileges that cannot be guaranteed by ETL Manager. If security rights are valid, use a pre-load procedure to issue the desired SQL or to run a user program that removes the rows using the most efficient method for your relational data target system.

Commit transactions every ____ rows

Specifies how often to commit a transaction. In general, a higher value improves performance while a lower value improves integrity. A value of zero (0) does not commit rows until all rows have been loaded.

Prior to Load

Specifies whether to remove data prior to loading the data target. This section does not exist for a flat file. The options are:

Leave Table does not delete the rows already in a data target.

Drop Table drops and recreates the data target. This option is available only for FOCUS and Fusion data targets.

Delete rows from table deletes all rows prior to loading data. A DELETE FROM TABLE command is issued against the target, and a log of the deleted records is created so that a Rollback command can later be issued to recover them. This value is not available for FOCUS and Fusion data targets.

Truncate logically empties the table. A log is not created, so this is often faster than selecting *Delete rows from table*. However it requires privileges that may not be in effect, so if the Truncate command fails, the result is as if you had chosen to delete the rows. This value is not available for FOCUS and Fusion data targets.

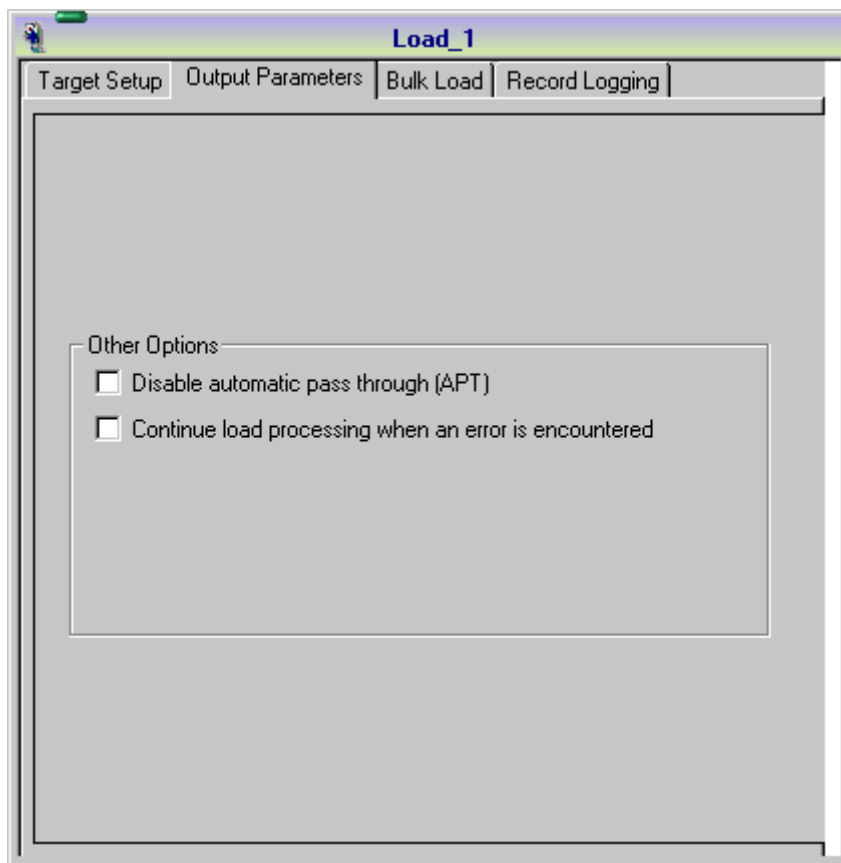
Setting Up Output Parameters

You must select output parameters for your data target if you are using variables in your ETL Request. The output parameters you must specify vary, depending on the data target type you selected.

- If you are using a relational data target, see *How to Set Output Parameters for a Relational Data Target* on page 4-83.
- If you are using a non-relational data target, see *Selecting a Data Target* on page 4-46.
- If you are using a flat file, see *How to Set Output Parameters for a Flat File* on page 4-86.
- If you are using a Server Transfer File, see *Server Transfer Files* on page 4-47.

Procedure How to Set Output Parameters for a Relational Data Target

1. Click the *Output Parameters* tab in the Load window. The Output Parameters tab opens.



2. Select *Disable automatic pass through (APT)* if you want to disable APT. For information on ATP, see Chapter 13, *Improving Performance*.
3. Select *Continue load processing when an error is encountered* to continue ETL processing if an error occurs.

By default, a load procedure to an RDBMS table is ended if a severe error is generated by the RDBMS for a particular row. Sometimes you want these errors to be ignored, especially when you anticipate duplicates in your input data or when you are including special “triggers” in your application. It is often advisable under these conditions to write your own input logic because error control can be complicated.

This is not available when using an Oracle target.

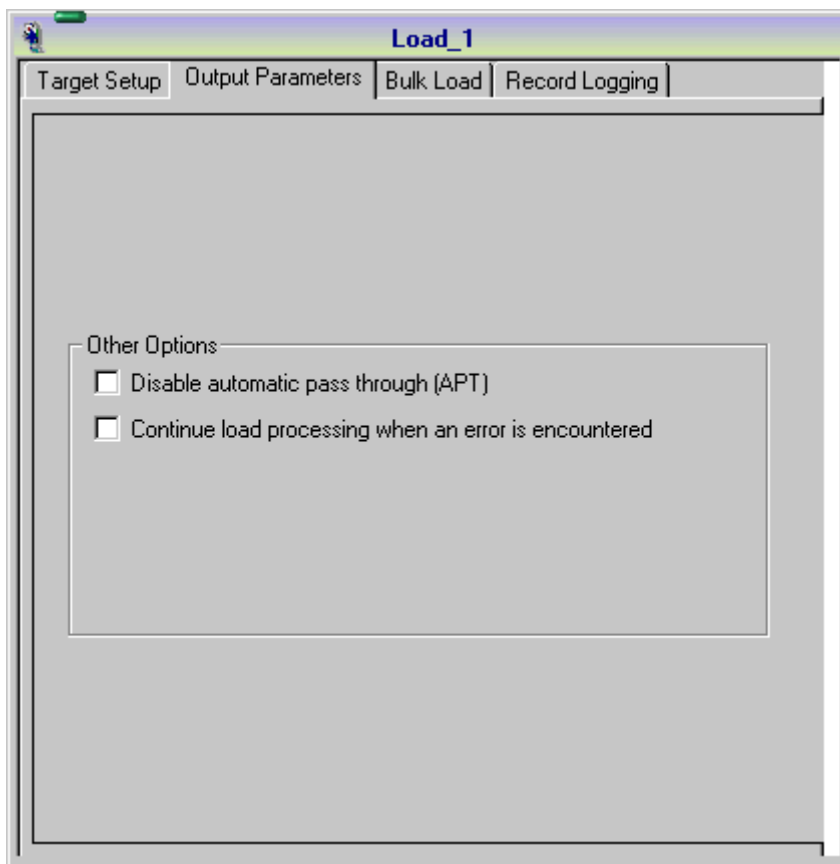
4. If you are using an Oracle data target, select a setting for Oracle Blank Column from the drop-down list.

VAR uses variable columns.

FIX uses fixed columns.

Procedure How to Set Output Parameters for a Non-Relational Data Target

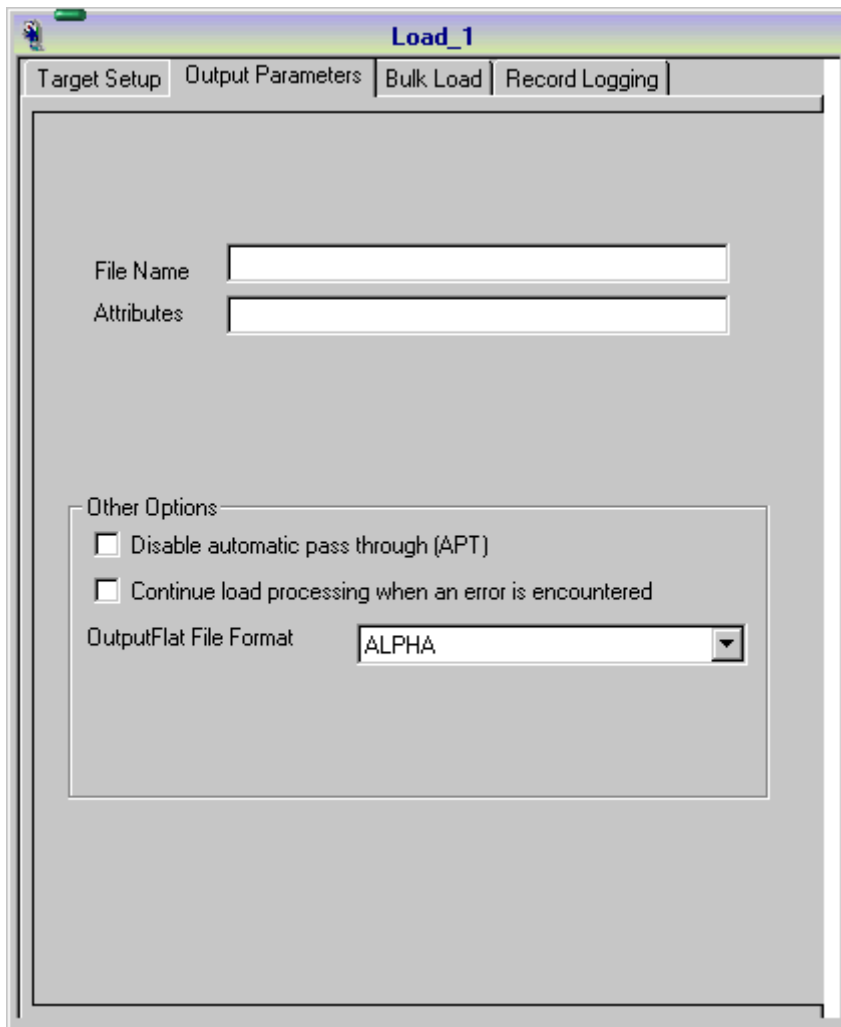
1. Click the *Output Parameters* tab in the Load window. The Output Parameters tab opens.



2. Select *Disable automatic pass through (APT)* to avoid using ATP. For information on ATP, see Chapter 13, *Improving Performance*.
3. Optionally, select *Continue load processing when an error is encountered* to avoid stopping the transfer when an error occurs.

Procedure **How to Set Output Parameters for a Flat File**

1. Click the *Output Parameters* tab in the Load window. The Output Parameters tab opens.



The screenshot shows a window titled "Load_1" with four tabs: "Target Setup", "Output Parameters", "Bulk Load", and "Record Logging". The "Output Parameters" tab is selected. Inside the tab, there are two text input fields labeled "File Name" and "Attributes". Below these fields is a section titled "Other Options" which contains two checkboxes: "Disable automatic pass through (APT)" and "Continue load processing when an error is encountered". At the bottom of the "Other Options" section is a dropdown menu labeled "OutputFlat File Format" with "ALPHA" selected.

2. Enter the path and file name for the subserver operating system in the File Name field.
3. If you are using OS/390, enter the attributes to allocate to the data set in the Attributes field.
4. Optionally, select *Disable automatic pass through (APT)*. For information on ATP, see Chapter 13, *Improving Performance*.

5. Optionally, select *Continue load processing when an error is encountered* to avoid stopping the transfer when an error occurs.
6. Select an output format for the flat file from the Output Flat File Format drop-down list. The options are:

Alpha produces a character file. All numeric data types are converted to characters. This format is the default.

Comma creates a comma-delimited file. This is useful for load programs that use this format.

Tabs creates a tab-delimited file.

Binary creates a file using the internal representation of numeric data types. All fields are aligned a full word (4 byte) boundary. Note that each field will start on a full word, so if you have a 1 byte field, it will skip 3 bytes before writing the next one.

Internal creates a file using the internal representation of numeric data types with no alignment.

Lotus creates a Lotus PRN file that can be imported into spreadsheet programs.

Comt creates a comma-delimited file with a header row containing the column names with trailing blanks removed from column values. This requires a 5.1.0 Server or later.

XML creates an XML document.

Modify creates a flat file using MODIFY instead of HOLD.

Procedure How to Set Output Parameters for a Server Transfer File

1. Click the *Output Parameters* tab in the Load window. The Output Parameters tab opens.

The screenshot shows a window titled "Load_1" with four tabs: "Target Setup", "Output Parameters", "Bulk Load", and "Record Logging". The "Output Parameters" tab is selected. The main area contains several input fields and a group box. The fields are: "File Name" (text box), "Attributes" (text box), "Operating System" (drop-down menu), and "Server" (drop-down menu). Below these is a group box titled "Other Options" containing two checkboxes: "Disable automatic pass through (APT)" and "Continue load processing when an error is encountered". At the bottom of the group box is a field labeled "OutputFlat File Format" with a drop-down menu showing "ALPHA".

2. Enter the path and file name for the subserver operating system in the File Name field.
3. If you are using OS/390, enter the attributes to allocate the data set in the Attributes field.
4. Select the operating system on which the target table resides from the Operating System drop-down list.
5. Select the server the target is located on from the Server drop-down list.

6. Optionally, select *Disable automatic pass through (APT)*. For information on ATP, see Chapter 13, *Improving Performance*.
7. Optionally, select *Continue load processing when an error is encountered* to avoid stopping the transfer when an error occurs.
8. Select an output format for the flat file from the Output Flat File Format drop-down list. The options are:

Alpha produces a character file. All numeric data types are converted to characters. This format is the default.

Comma creates a file in comma delimited file. This is useful for load programs that use this format.

Tabs creates a tab delimited file.

Binary creates a file using the internal representation of numeric data types. All fields are aligned a full word (4 byte) boundary. Note that each field will start on a full word, so if you have a 1 byte field, it will skip 3 bytes before writing the next one.

Internal creates a file using the internal representation of numeric data types with no alignment.

Lotus creates a Lotus PRN file that can be imported into spreadsheet programs.

Comt creates a comma-delimited file with a header row containing the column names with trailing blanks removed from column values. This requires a 5.1.0 Server or later.

Using Bulk Loading

You can use bulk loading to process your data. This is specified by selecting the Use alternate load procedure checkbox in the Target Setup tab. Bulk loading allows you to specify the target details for the type of file you have selected to load. iWay is still used to extract your data from the data target. However, instead of using iWay to extract your data from the data target and insert it into the data target you can use bulk loaders to insert data into the data target. ETL Manager automates bulk loading for Ingres Informix, Microsoft SQL Server, IBM DB2, NCR Teradata, Nucleus, Oracle, and Sybase.

Note: Bulk loading is only available for relational data targets.

When using bulk loading, you must specify bulk load details. You must specify the following:

- Information about the data target. For details, see *Specifying Target Information for Bulk Loading* on page 4-90.
- Alternate files. In most cases you will use bulk load without any additional specifications. However, in some cases they may be necessary. For instructions on how to specify these files, see *Specifying Alternate Files for Bulk Loading* on page 4-92.
- Run options. For details, see *Selecting Run Options* on page 4-93.
- If you are using a Oracle, DB2, Nucleus, or Teradata target, additional options. For details, see *Setting Additional Options for Oracle, DB2, Nucleus, and Teradata* on page 4-95.

Reference Prerequisites for Using Bulk Loading

In order to use ETL Manager to control the bulk loading procedure, you need parameters including the name and owner of the table where you are inserting data, the user ID and password to log into the RDBMS. You supply this information in the ETL Workbench.

ETL Manager will automatically generate:

- An extract file containing the data to insert. This can be a flat file or a Server Transfer File generated by ETL Manager, or a file created in another way.

Note: If the file has been created outside of ETL Manager, you need a Synonym.

See the *iWay Server Administration Version 5 Release 1.0* for more information.

- A control file describing the data being inserted.

Specifying Target Information for Bulk Loading

The first step in using bulk loading is to specify target information.

Procedure How to Specify Target Information for Bulk Loading

1. Click the *Bulk Load* tab. The Bulk Load tab opens with the Target Database tab open.

The screenshot shows a window titled "Load_1" with a tabbed interface. The main tabs are "Target Setup", "Output Parameters", "Bulk Load", and "Record Logging". The "Bulk Load" tab is selected. Inside the "Bulk Load" tab, there are sub-tabs: "Target Database", "Alternate Files", and "Run Options". The "Target Database" sub-tab is selected. The main area of the "Target Database" sub-tab contains four labeled text input fields: "User ID", "Password", "Database", and "Owner".

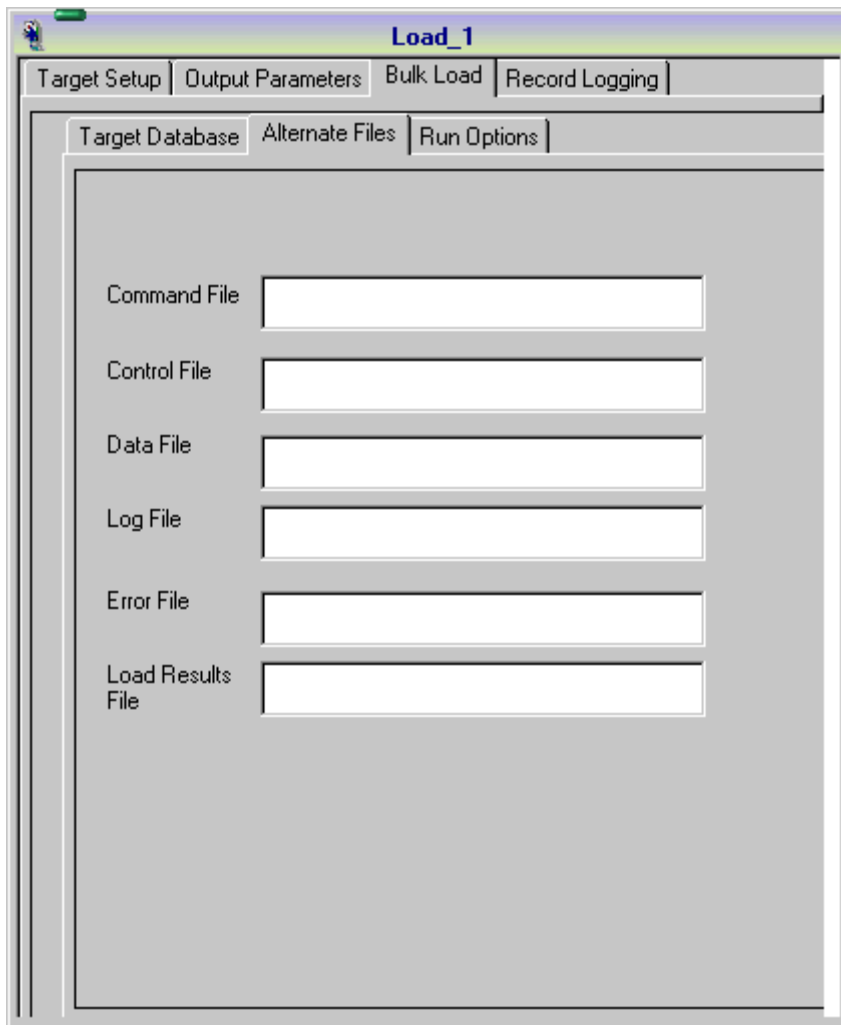
2. Enter a valid user ID in the User ID field.
3. Enter the password for the user ID in the Password field.
4. Enter the data target in the Database field.
5. Enter the owner in the Owner field.

Specifying Alternate Files for Bulk Loading

You can select user-created control files instead of the files automatically created by ETL Manager during bulk loading. You can create an alternate command file, control file, data file, log file, and error file.

Procedure How to Select Alternate Files

1. Click the *Alternate Files* tab in the Bulk Load tab. The Alternate Files tab opens.



The screenshot shows a window titled "Load_1" with a tabbed interface. The main tabs are "Target Setup", "Output Parameters", "Bulk Load", and "Record Logging". The "Bulk Load" tab is selected. Within the "Bulk Load" tab, there are sub-tabs: "Target Database", "Alternate Files", and "Run Options". The "Alternate Files" sub-tab is active. The main area of the dialog contains six labeled text input fields arranged vertically: "Command File", "Control File", "Data File", "Log File", "Error File", and "Load Results File". All fields are currently empty.

2. Enter the full path and file name for the command file you want created in the Command File field. The default is cmblkcmd.bat.

3. Enter the full path and name for the control file you want created in the Control File field. The default is cmbkctl.ftm.
4. Enter the full path and name for the data file you want created in the Data File field. The default is cmsqlget.ftm.
5. Enter the full path and name for the log file you want created in the Log File field. The default is cmbklog.ftm.
6. Enter the full path and name for the error file you want created in the Error File field. The default is cmbkout.ftm.
7. Enter the full path and name for the load results file you want created in the Load Results File field.

Selecting Run Options

You can select run options for your bulk load process.

Procedure How to Select Run Options

1. Select the *Run Options* tab in the Bulk load tab. The Run Options tab opens.

The screenshot shows a dialog box titled "Load_1" with a tabbed interface. The "Bulk Load" tab is selected, and within it, the "Run Options" sub-tab is active. The "Run Options" sub-tab contains several settings: two unchecked checkboxes labeled "Drop table prior to load" and "Stop after control and command file created"; a "Data Path" text field; and five rows of settings, each with a text field and a label: "Packet size" (bytes), "Maximum errors", "Skip" (records before starting), "Load" (records before stopping), and "Version Number".

2. Select *Delete table prior to load* to delete the data target before loading new data into it. This is the default.
3. Select *Stop after control and command file created* to stop the bulk load processing after the command and control file are created.

You can use this parameter to make further changes to either the command file or the control file after ETL Manager creates them, but before the data is loaded into the data target.

This option enables a command and control file to be built, but does not load data. Setting it stops the bulk load after the command and control files are created. The names of the command and control files are `cmbkcmd.bat` and `cmbkctl.ftm`, respectively.

4. Enter the location of the data command and control files used in conjunction with alternate transfer local directories in the Data Path field. Include a slash at the end of the path.
5. Enter the packet size in bytes in the Packet size field.
Note: This option is not applicable for Informix.
6. Enter the maximum number of errors allowed in the Maximum errors field. The default is the target RDBMS's default.
7. Enter the number of rows to skip before loading in the Skip _____ records before starting field.
8. Enter the number of records to load, not including skipped rows, in the field.
9. Enter the version number in the Version Number _____ field. This parameter's effect depends on the data target type.
10. For Sybase, SQL Server, or Oracle, enter the version number of the vendor's bulk load utility.

For Sybase, the default is 10.0. For SQL Server, the default is 7.0.

For Oracle, specify as

`SQLLDRnn`

where:

`nn`

Is one of the following:

`sqlldr70` for Oracle 7.0.

`sqlldr72` for Oracle 7.2.

`sqlldr80` for Oracle 8.0.

`sqlldr` for Oracle 8.1 or Oracle 8i.

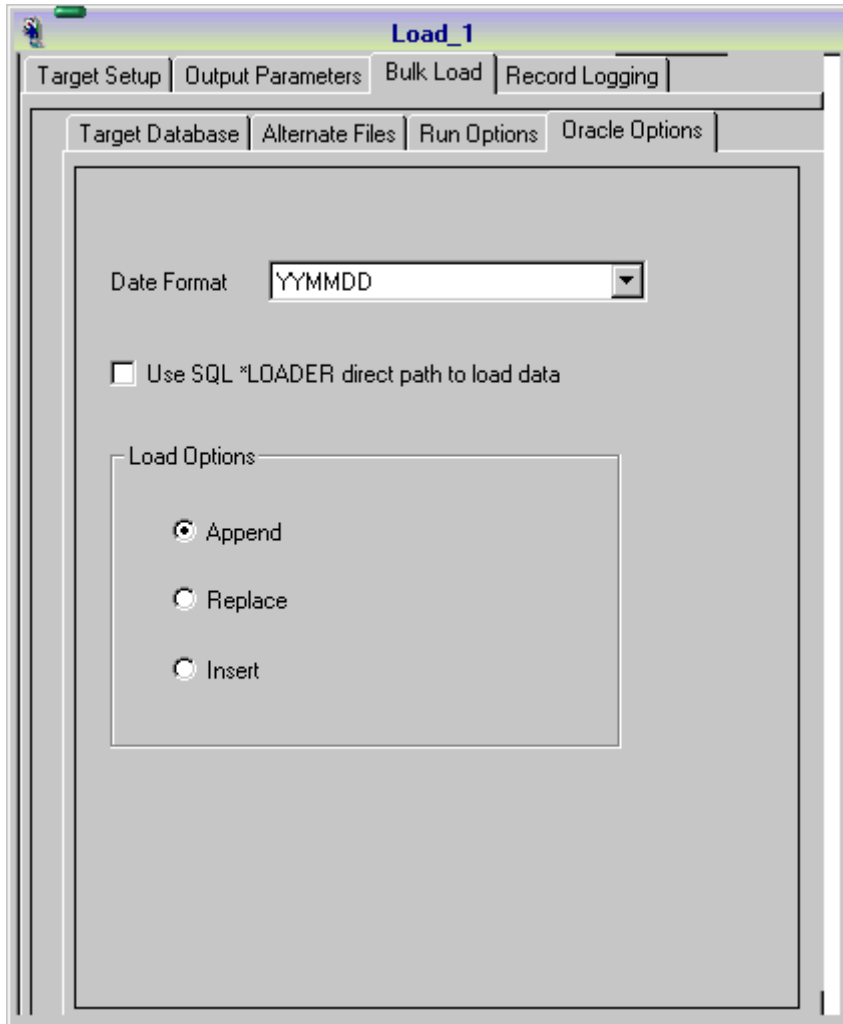
Once ETL Manager has all of this information, it can perform the bulk loading as part of the ETL Request process.

Setting Additional Options for Oracle, DB2, Nucleus, and Teradata

When you use bulk loading with an Oracle, DB2, Nucleus, or Teradata data target you must set options for your data target that are specific to the data target type.

Procedure How to Set Additional Options for Oracle

1. Click the *Oracle Options* tab. The Oracle Options tab opens.



2. Select a date format from the Date Format drop-down list. Your options include:
YYMMDD loads dates in year-month-day format.
MMDDYY loads dates in month-day-year format.

3. Optionally, select *Use SQL*LOADER direct path to load data* to use the direct path load option of SQL*LOADER. This avoids multiple inserts by creating and using pre-formatted data blocks. Indexes are placed in a Direct Load state. After the load is complete, Oracle merges the old index with the new one.

4. Select a load option:

Append adds rows to the existing table. This is the default.

Replace empties the table and then adds rows. The user must have delete privileges to use this option.

Insert adds rows to an existing table or otherwise aborts.

Procedure How to Set Additional Options for DB2

1. Select the DB2 Options tab. The DB2 Options tab opens.

The screenshot shows the 'Load_1' dialog box with the 'DB2 Options' tab selected. The dialog has a title bar 'Load_1' and a tabbed interface with tabs: 'Target Setup', 'Output Parameters', 'Transfer Details', 'Bulk Load', and 'Record'. The 'DB2 Options' tab is active, showing a sub-dialog with tabs: 'Target Database', 'Alternate Files', 'Run Options', and 'DB2 Options'. The 'DB2 Options' sub-tab contains the following fields:

- Company Identifier: [Text Field]
- High Level Qualifier For ETLManager Data Sets: [Text Field]
- Job Card Information:
 - Job Name: [Text Field]
 - Accounting Information: [Text Field]
 - Name: [Text Field]
 - MSGLevel: [Text Field]
 - Notify: [Text Field]
 - MSGClass: [Text Field]
 - Class: [Text Field]
- ETLManager Command File: [Text Field]
- ETLManager Control File: [Text Field]
- ETLManager Data File: [Text Field]
- ETLManager Discarded Data File: [Text Field]

- 2.** Optionally, enter a high level qualifier for the data set name in the Company Identifier field.
- 3.** Enter the job card information in the following fields:
 - a.** Enter the name of the job in the Job Name field.
 - b.** Enter the name the name that will identify the run in the Name field.
 - c.** Enter the job class in the Job Class field.
 - d.** Enter the job scheduler output class in the MSGCLASS field.
 - e.** Specify whether or not to list the JCL statement in the MSGLEVEL field. The default entry is 1,1.
 - f.** Enter a user to be notified at the completion of the job in the Notify field.
- 4.** Optionally, enter a fully-qualified file name for the command file in the CM Command File field.
- 5.** Optionally, enter a fully-qualified file name for the control file in the CM Control File field.
- 6.** Optionally, enter a fully-qualified file name for the data file in the CM Data File field.
- 7.** Optionally, enter a fully-qualified file name for the discarded data file where DB2 will place discarded records in the CM Discarded Data File field.

Procedure How to Set Additional Options for Nucleus

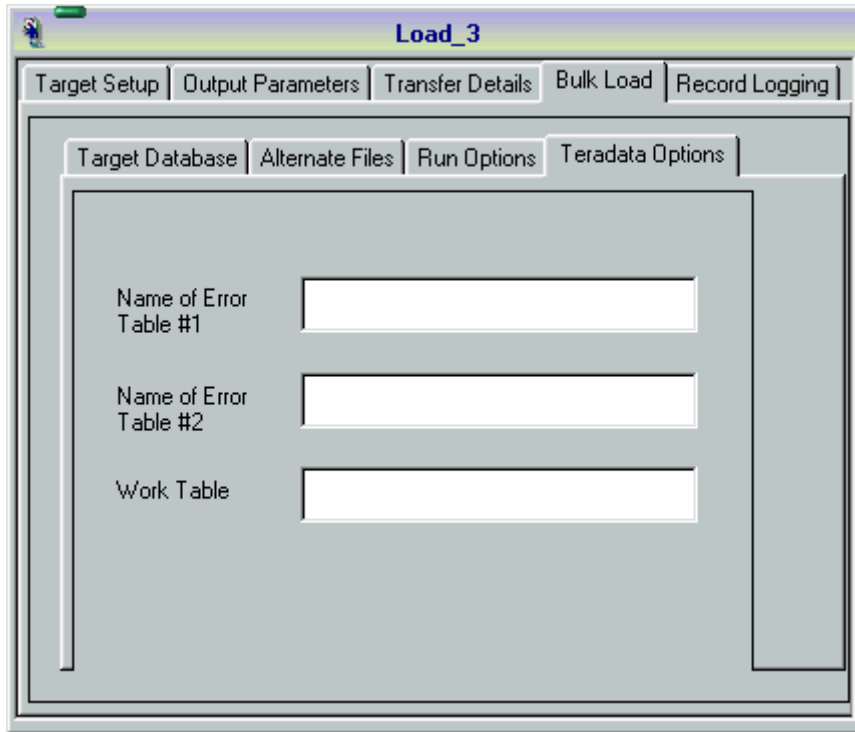
1. Click the *Nucleus Options* tab. The Nucleus Options tab opens.

The screenshot shows a window titled "Load_3" with a tabbed interface. The tabs are "Target Setup", "Output Parameters", "Transfer Details", "Bulk Load", and "Record Logging". The "Nucleus Options" tab is selected. Inside this tab, there are sub-tabs: "Target Database", "Alternate Files", "Run Options", and "Nucleus Options". The "Nucleus Options" sub-tab is active, showing a "Server(Host) Name" text field and two unchecked checkboxes: "Commit loading row processing if any error occurs" and "Reject unprintable characters".

2. Enter the server name in the Server(Host) Name field. The server name determines the name of the system the Nucleus server is running on.
3. Optionally, select *Commit loading row processing if an error occurs* to commit rows that have already been loaded if an error occurs.
If you do not select this option and an error occurs, no rows will be committed.
4. Optionally, select *Reject unprintable characters* to reject unprintable characters.

Procedure **How to Set Additional Options for Teradata**

1. Click the *Teradata Options* tab. The Teradata Options tab opens.



The screenshot shows a window titled "Load_3" with several tabs: "Target Setup", "Output Parameters", "Transfer Details", "Bulk Load", and "Record Logging". The "Teradata Options" tab is selected. Inside this tab, there are three input fields: "Name of Error Table #1", "Name of Error Table #2", and "Work Table". Each field is currently empty.

2. Enter the name of the error table #1 in the Name of Error Table #1 field.
This is the table that contains constraint violations, conversion errors, and unavailable AMP conditions.
3. Enter the name of the error table #2 in the Name of Error Table #2 field.
This is the table that contains unique primary index violations.
4. If you are using the MLOAD utility, enter the work table name in the Work Table field.
Work Table is a special unhashed table used by MLOAD utility.

Note: Error Table #1 and Error Table #2 are the two tables where the FastLoad and MLOAD utilities store the input records related to constraint violations, conversion errors, unavailable AMP conditions, and unique primary index violations.

Using Bulk Load Command and Control Files

ETL Manager generates bulk load command and control files for standard ETL Request processing. In some cases it may be useful to modify the command and control files for specific needs. This requires knowledge of the relational data target bulk load program syntax. In a two step procedure, the files are created and the request is run after the files are modified.

Procedure How to Create the First ETL Request

1. Create your ETL Request, including a filter to reduce the amount of data.
2. In the Target window, click the *Bulk Load* tab.
3. Enter a user ID and password in the User ID and Password fields.
4. Click the *Run Options* tab.
5. Enter the file path for the command and control files in the Data Path field. Be sure to include a backslash at the end of the path.

ETL Manager will write the bulk command and control files to this directory. If it is not specified, the command and control files will be written to iWay's temporary working directory and deleted after the process completes.

6. Select *Stop after command and control file created*.
7. Execute the ETL Request.
8. Edit the CMBLKCTL.FTM file specified in the Bulk Datapath directory (this is your control file). For information on control files, please consult the native RDBMS bulk loading instructions.

You have created your custom control file; you are now ready to create the second ETL Request.

Procedure How to Create the Second ETL Request

1. Make a copy of your original ETL Request.
2. Remove the filter condition from your ETL Request.
3. Unselect *Stop after command and control file created*.
4. Click the *Alternate Files* tab.
5. Enter the location and name of your control file in the Data Path field.
6. Re-execute your ETL Request.

Logging Records

Transactions are written to ETL Manager's log table, which can be viewed from the ETL Workbench. However, it may be desirable to log the records to a separate file.

The following types of transactions can be logged:

- All transactions.
- Accepted transactions.
- Transactions rejected due to a duplicate key.
- Transactions rejected when a required key is not found.
- Invalid data, or data rejected by RDBMS.
- Format error.

Procedure How to Specify Logging Options

1. In the Load window, click the *Logging Records* tab. The Logging Options tab opens.

The screenshot shows a window titled "Load_1" with four tabs: "Target Setup", "Output Parameters", "Bulk Load", and "Record Logging". The "Record Logging" tab is selected. Inside the tab, there is a "Master File Description Name:" label followed by a text input field. Below this, there is a table with two columns: "File Name" and "Attributes". The table has six rows, each corresponding to a transaction type with a checkbox on the left:

	File Name	Attributes
<input type="checkbox"/> All Transactions		
<input type="checkbox"/> Accepted Transactions		
<input type="checkbox"/> Duplicate Rejections		
<input type="checkbox"/> No Match Rejections		
<input type="checkbox"/> Invalid Data		
<input type="checkbox"/> Format Error		

2. Enter the file name of the Master File for the log file in the Master File Description Name field.
3. Select the check box next to the type or types of transactions you want logged.
4. For each transaction type you select, enter the path and file name to which the transactions will be logged in the File Name field.
5. For OS/390, for each transaction type you selected, enter the attributes for the log file in the Attributes field.

CHAPTER 5

Scheduling and Executing ETL Requests

Topics:

- Scheduling an ETL Request
- Restarting a Failed ETL Request
- Creating Dependencies
- Event-Based Scheduling

After building an ETL Request, you can specify an execution schedule.

- The Scheduler tool enables you to specify a starting date and time on which to execute your ETL Request.
- The Dependencies tool enables you to order additional ETL Requests to run after a request completes.

You can also use event-based scheduling, which starts ETL Requests from other platforms and processes.

Scheduling an ETL Request

You can schedule the execution of your ETL Requests with the ETL Manager Scheduler. For details on the Scheduler tool, see *Scheduler Window* on page 5-2.

You can schedule a query to run in the following ways:

- **Run Once.** Runs your query once.
- **Recurring.** Repeats your query over an interval that you define.
- **Multi-Day.** Runs your query on certain days of the month.

The Scheduler checks for scheduled ETL Requests at intervals specified in the Web Console. When the Scheduler checks for ETL Requests, it executes all ETL Requests scheduled to run in the period of time since the Scheduler last checked for ETL Requests.

You can also execute an ETL Request immediately. For information, see *Executing an ETL Request Immediately* in Chapter 7, *Working With an ETL Request*.

Reference Scheduler Window

The screenshot shows the ETL Manager Scheduler window with the following sections:

- Schedule Type:** Three radio buttons: ☐ Run Once, ☒ Recurring, and ☐ Multi-Days.
- Interval Type:** Six radio buttons: ☐ Minutes, ☒ Hourly, ☐ Daily, ☐ Weekly, ☐ Monthly, and ☐ Yearly.
- Interval Number:** A text box containing '01' with up and down arrow buttons.
- Start Time:** Fields for HH (10), MM (15), DD (26), MM (11), and YYYY (2002), followed by a calendar icon.
- End Time:** Fields for HH (00), MM (00), DD, MM, and YYYY, followed by a calendar icon.
- Days of the Week:** A row of seven buttons: Sun, Mon, Tue, Wed, Thu, Fri, Sat.
- Days of the Month:** A 5x7 grid of buttons for days 1 through 31. The last cell contains a checkbox labeled 'Last Day of Month'.

The Scheduler window contains the following fields/options:

Schedule Type

Determines whether a request will execute once, on a recurring basis, or on several specific days.

Interval Type

Specifies the interval in which a request will be executed. For example, if you select Month as your Interval Type, and select 2 as your Interval Number, the ETL Request will execute every two months.

Interval Number

Specifies the frequency at which a request will be executed. For example, if you select Week as your Interval Type, and select 2 as your Interval Number, the ETL Request will execute every two weeks.

Start Time

Specifies the start time and date for the execution of the request.

End Time

Specifies the end time and date for the execution of the request. This is optional.

Days of the Week

Specifies the day of the week on which the request will be executed. Depending on the selection made in the Interval section, this may be grayed out.

Days of the Month

Specifies the day of the month on which the request will be executed. Depending on the selection made in the Interval section, this may be grayed out.

Procedure How to Schedule an ETL Request to Run Once

1. In the Scheduler window, select *Run Once* in the Schedule Type section.
2. Select a start time in the Start Time section by entering the time in the fields, or using the arrow buttons.
3. Select a start date in the Start Time section by entering the date, or by clicking the calendar icon and choosing a date from the Calendar window that opens.

Procedure How to Schedule an ETL Request to Run on a Recurring Interval

1. In the Scheduler window, select *Recurring* in the Schedule Type section.
2. Select an interval type from the Interval Type section.
3. Select an interval number from the Interval Number section.

4. Select a start time in the Start Time section by entering the time in the fields, or using the arrow buttons.
5. Select a start date in the Start Time section by entering the date, or by clicking the calendar icon and choosing a date from the Calendar window that opens.
6. Optionally, select an end time in the End Time section by entering the time in the fields, or using the arrow buttons.
7. Optionally, select an end date in the End Time section by entering the date, or by clicking the calendar icon and choosing a date from the Calendar window that opens.

Note: You do not need to change the schedule to stop a query from executing regularly. Instead, set the status of the request to Inactive in the ETL Request main window. For details on setting the status of an ETL Request see Chapter 7, *Working With an ETL Request*.

Procedure **How to Schedule an ETL Request to Run at Multi-Day Intervals**

1. In the Scheduler window, select *Multi-Days* in the Schedule Type section.
2. Select the days of the week or month on which you want your job run from the Days of the Week or Days of the Month section.

If you want your job to run at the end of the month, select Last Day of Month.

3. Select a start time in the Start Time section by entering the time in the fields, or using the arrow buttons.
4. Select a start date in the Start Time section by entering the date, or by clicking the calendar icon and choosing a date from the Calendar window that opens.
5. Optionally, select an end time in the End Time section by entering the time in the fields, or using the arrow buttons.
6. Optionally, select an end date in the End Time section by entering the date, or by clicking the calendar icon and choosing a date from the Calendar window that opens.

Note: You do not need to change the schedule to stop a query from executing regularly. Instead, set the status of the request to Inactive in the ETL Request main window. For details on setting the status of an ETL Request see Chapter 7, *Working With an ETL Request*.

Restarting a Failed ETL Request

When an ETL Request's execution fails, you can control whether or not the execution is attempted again with the Restart/Recovery facility. You can set the Restart/Recovery facility for an individual request or for all requests.

Procedure How to Specify a Request to Restart

1. In the Restart/Recovery section of the ETL Request main window, select the number of times to restart your request from the Number of Attempts spin box.
2. Select where to restart the ETL Request from in the Restart from drop-down list. The options are:
 - *Beginning* restarts the request at the first record.
 - *Last Commit* restarts the request after the last commit point. This option is only effective for target data.

Creating Dependencies

The Dependencies tool enables you to schedule ETL Requests to execute in a specific order. Use this feature to schedule an ETL Request to run immediately upon the successful completion of another ETL Request. If an ETL Request does not finish executing properly and returns any error messages, dependent ETL Requests will not execute.

For example, if you are creating an employee information data mart, an ETL Request loads data from the personnel database, and another ETL Request loads data from the accounting database. You need to make sure that the second ETL Request does not run until the first ETL Request completes. Therefore, you would schedule the first ETL Request and schedule the second ETL Request to run when the first ETL Request is complete. This eliminates trying to estimate what time the first ETL Request will end and scheduling the second ETL Request to run at that time.

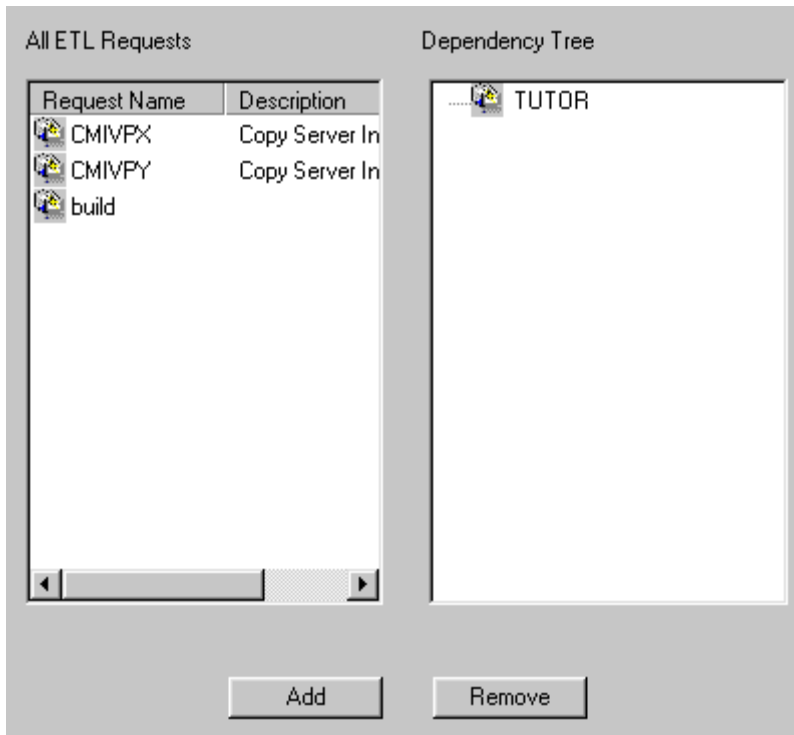
For details on the Dependencies tool, see *Dependencies Window* on page 5-6.

Reference Guidelines for Scheduling Dependent ETL Requests

The following guidelines apply when scheduling dependent ETL Requests:

- You can run up to ten dependent requests.
- The ETL Manager log for the request displays only the first dependent request.

Reference Dependencies Window



The Dependencies window contains the following fields/options:

All ETL Requests

Lists all the available ETL Requests.

Request Name

Is the name of the ETL Request.

Description

Is a description of the ETL Request, if available.

Dependency Tree

Lists the first ETL Request to execute, and all the ETL Requests scheduled to run dependent on its execution.

Add

Adds an ETL Request to the Dependency Tree.

Remove

Removes an ETL Request from the Dependency Tree.

Procedure How to Schedule a Dependent ETL Request

1. In the ETL Workbench main window, open the ETL Request you wish to run.
2. Click the *Dependencies* tool.
3. Select the ETL Request you want to run after the completion of the first request from the All ETL Requests list.
4. Click *Add*, or drag the request to the Dependency Tree list.
5. Repeat steps three and four for each ETL Request you want to run.

To change the order in which a dependent request executes, drag the request up or down within the Dependency Tree list to where you want it to be in the order of execution.

Procedure How to Remove a Dependent ETL Request

1. Select the request you wish to remove.
2. Click *Remove*, or drag the request to the All ETL Requests list.

Event-Based Scheduling

You can perform event-based scheduling, which executes an ETL Request from an application other than the ETL Workbench. Event-based scheduling can be useful when you want a remote application to run an ETL Request. For example, you might have a Visual Basic application that requires an ETL Request to run. By adding a button to your application, you could send a request to the Server to start an ETL Request.

Event-based scheduling can also be useful when performing a job that updates your data. At the end of this update you can send a message to the Server to run an ETL Request to ensure the ETL Request will be using the most updated data. For example, when running a batch job on OS/390, you can make the final step in the batch job sending a message to the Server telling it to execute the ETL Request. In this instance, the ETL Request is guaranteed to be accessing the most updated data.

ETL Manager allows you to perform event-based scheduling with the following:

- CMRUN is a stand-alone program that initiates ETL Requests. It is available for OS/390, VM, UNIX, and Windows Me/NT/2000. For details, see *Starting ETL Requests With CMRUN* on page 5-8.
- CMASAP enables event execution from an API program or from FOCUS. For details, see *Specifying an External Password* on page 5-13.
- An API program you write can invoke the ETL Request from a remote platform. For details, see *Starting ETL Requests With Your Own API Program* on page 5-15.
- A FOCUS client using specific FOCUS code can initiate an ETL Request. For details, see *Starting ETL Requests From FOCUS* on page 5-17.

Note: Using CMRUN is the preferred method of performing event-based scheduling, and should be used whenever possible.

Starting ETL Requests With CMRUN

The CMRUN program is a standalone program for initiating ETL Requests. It is provided with ETL Manager for OS/390, most UNIX systems, and Windows Me/NT/2000. It may be called from an OS/390 job stream, a UNIX shell script, or a Windows shortcut or command file.

After an ETL Request is executed with CMRUN, a return code is returned alerting the user of whether the ETL Request executed successfully. For details on these return codes, see *CMRUN Return Codes* on page 5-12.

CMRUN is called from the ETL Manager installation directory, which is by default:

`drive:\Program Files\iWay\Data Management Administration Tools 5.2\ETL Manager\cmrun.exe`

Before calling CMRUN on Windows NT/2000/XP, you must set up the environment. The following are examples using the default file locations:

```
set EDAHOME=C:\ibi\client52\home
set EDACONF=C:\ibi\client52\cln
set EDACS3=C:\ibi\client52\cln\etc\odin.cfg
set PATH=C:\ibi\client52\home\bin;%PATH%
```

Note: An iWay communication file (odin.cfg) must be configured for the Server using the iWay Connector configuration to use CMRUN.

Syntax**How to Start ETL Requests With CMRUN**

```
CMRUN Server=server, [User=userid, [Password=password,] ] [,x=ddname]
,Request=request [,Appname=appname] [,Wait=minutes] [,I=seconds]
[,B=begin] [,E=end]
```

where:

server

Is the name of the Server. This value is required.

userid

Is a valid user ID for the server. It is required for a secured server.

password

Is the password associated with the user ID. This is required for a secured server.

ddname

Is the location of the external password.

request

Is the name of the ETL Request, up to eight characters long.

appname

Is the application in which the ETL Request is located.

minutes

Is the number of minutes the server waits for an ETL Request to complete. The default is 0, which specifies not to wait.

seconds

Is the number of seconds the server waits before checking if the ETL Request has completed. The default is 15 seconds.

begin

Is the record to start loading at. This is not available for flat files or Server Transfer Files.

end

Is the maximum number of records to retrieve from the Server.

Note: Parameters may be abbreviated as its initial letter. Parameters must be separated by a comma and space.

Example **Sample CMRUN Batch File for Windows Me/NT/2000**

The following is an example of a Windows batch file that calls CMRUN:

```
cmrun s=chkserve, u=edachk, p=pass, r=test, w=5
if errorlevel 20 goto rc20
if errorlevel 16 goto rc16
if errorlevel 12 goto rc12
if errorlevel 8 goto rc8
if errorlevel 4 goto rc4
echo Request OK
goto end
:rc20
    echo (20) Request timed out
    goto end
:rc16
    echo (16) Request failed in execution
    goto end
:rc12
    echo (12) Request not found
    goto end
:rc8
    echo (8) iWAY API Error
    goto end
:rc4
    echo (4) Invalid input parameter
:end
```

Example **Running an ETL Request Using CMRUN Without Waiting for Completion**

In the following example, CMRUN is called and ETL Manager does not wait for the procedure to complete:

```
Run ( 1) Connecting to 'demo' as 'etl'
Run ( 2) ETL Manager Server: Version 5, Release 2
ICM (18774) Deferred request decode created
ICM (18762) ETL Job ID: 2002-11-13-18-37-28cmrpi000007_loopback
```

Example **Running an ETL Request Using CMRUN and Waiting for Completion**

In the following example, CMRUN is called and ETL Manager waits for the procedure to complete. The procedure is executed successfully.

```
Run ( 1) Connecting to 'demo' as 'etl'
ETL Manager Server: Version 5, Release 2
```

At completion, the following messages display

```
ICM (18762) ETL Job ID: 2002-11-14-09-41-29t3rp3648_00000001
ICM (18763) ETL Request demo complete
```


In the following example, CMRUN is called and ETL Manager waits for the procedure to complete. The request fails.

```
Run ( 1) Connecting to 'demo' as 'etl'
ETL Manager Server: Version 5, Release 2
```

When the request fails, the following messages display:

```
ICM (18762) ETL Job ID: 2002-11-13-18-37-28t3rp232_00000002
ICM (18764) ETL Request decode failed; RC = 350
```

Example Sample CMRUN Shell Script for UNIX

The following is an example of UNIX shell script that calls CMRUN:

```
#!/bin/ksh
export SHLIB_PATH=$EDAHOME/bin:/usr/lib:/lib
export EDACS3=./odin.cfg
./cmrun s=EDASERV, r=runtest, u=edachk, password=pass, w=10, b=2, e=5
```

Example Sample CMRUN Code for SUN Solaris

The following is an example of SUN Solaris code that calls CMRUN:

```
#!/bin/ksh
LD_LIBRARY_PATH=$EDAHOME/bin:/user/lib:/lib
export EDACS3=./odin.cfg
./cmrun s=EDASERV, r=runtest, u=edachk, password=pass, w=10, b=2, e=5
```

Example Sample CMRUN Code for AIX

The following is an example of AIX code that calls CMRUN:

```
#!/bin/ksh
export LIB_PATH=$EDAHOME/bin:/user/lib:/lib
export EDACS3=./odin.cfg
./cmrun s=EDASERV, r=runtest, u=edachk, password=pass, w=10, b=2, e=5
```

Example Sample CMRUN Batch File for OS/390

The following is an example of OS/390 JCL that calls CMRUN:

```
//CMRUN JOB 'CMRUN',MSGCLASS=A,CLASS=S
//          MSGLEVEL=(1,0),REGION=4096K
//*****
//CMASAP    EXEC PGM=CMRUN,
//          PARM='s=CMINT, u=edachk, p=pass, r=runtest, w=5',
//          REGION=4096K
//*****
//STEPLIB DD DSN=qualif.EDALIB.LOAD,DISP=SHR
//          DD DSN=qualif.EDASASC.LINKLIB,DISP=SHR
//SYSPRINT DD SYSOUT=*
//SYSOUT DD SYSOUT=*
//SYSABOUT DD SYSOUT=*
//SYSTEM DD SYSOUT=*
//STDOUT DD SYSOUT=*
//SYSUDUMP DD SYSOUT=*
//EDAOUT DD SYSOUT=*
//EDACS3 DD DSN=qualif.INSTALL.DATA(EDACSG),DISP=SHR
```

where:

qualif

Is a high-level qualifier for the EDA data sets.

Note: If LU6.2 is used for communications, a DD statement must be added to identify the LU pool. For example:

```
//LU6POOL DD DSN=qualif.EDACTL.DATA(LU6POOL),DISP=SHR
```

Reference CMRUN Return Codes

When an ETL Request is completed, CMRUN sets a return code that can be viewed in the calling environment. These codes are as follows:

Code	Description
0	The value of the 0 code depends on the Wait parameter setting: <ul style="list-style-type: none">• If set to zero, the request was <i>submitted</i> successfully.• If set to a number greater than zero, the request <i>completed</i> successfully.
4	Invalid input parameter.
8	iWay API returned an error.

Code	Description
12	Request submission failed because the ETL Request was not found, was incomplete, or the user was not authorized.
16	The request failed during execution. This return code may occur only when the Wait parameter is set to a number greater than zero.
20	The request timed out. This return code may occur only when the Wait parameter is set to a number greater than zero.

An ETL Request may fail during connection to the Server or while being run. For details on the problems that may cause a failed ETL Request see Appendix B, *Problems, Errors, and Troubleshooting*.

Specifying an External Password

On OS/390, CMRUN allows you to supply the server password from an external data set that can be RACF or Top Secret. An X parameter specifies the ddname of an external data set (which must then be allocated) that contains the password.

The data set containing the password should contain a single line with the password. The password is case sensitive and is terminated by the first blank space so the password cannot contain blanks. It should be allocated with an LRECL=80 and RECFM=FB.

Example Specifying an External Password With SYSIN

The following provides an external password.

```
//SYSIN DD DISP=SHR, DSN=cmrun.password.data
//CMASAP EXEC PGM=CMRUN, PARM=('S=server, U=userid, R=request,
X=SYSIN'), ...
```

Starting ETL Requests With CMASAP

The CMASAP procedure enables event-oriented execution from an API program or FOCUS, and is provided with ETL Manager. This procedure is located in the stored procedure directory or PDS for your Server. The files are:

- **OS/390:** *qualif.EDARPC.DATA(CMASAP)*.
- **UNIX and Windows 2000/NT/Me:** *cmasap.fex*, located in the ETLMGR directory in the Server directory.

Syntax **How to Start ETL Requests With CMASAP**

CMASAP

```
REQ_NAME=name[ ,STOPAT=stopat] [ ,STARTAT=startat] [ ,CM_SETERR={ON|OFF} ]  
[ ,CM_ASYNC={ON|OFF} ]
```

where:

name

Is the name of the ETL Request, up to eight characters long.

startat

Is the record to start loading at. This is not available for flat files or Server Transfer Files.

stopat

Is the maximum number of records (rows) to retrieve from the Server. RECORDLIMIT and READLIMIT are passed to the Server.

CM_SETERR

Determines the types of messages returned to the calling program. Valid values are:

ON

CMASAP sends a confirmation message and administrative messages that are produced as CMASAP prepares an ETL Request for execution. This is the default.

OFF

CMASAP sends only the confirmation or failure message.

When putting an event-oriented program into production, set CM_SETERR to OFF so the calling program only needs to check one message when the Server processes the ETL Request.

CM_ASYNC

Determines if the server waits for the ETL Request to complete. Valid values are:

ON

CMASAP relinquishes control to the Server and terminates when the request is submitted. This is the default.

OFF

CMASAP waits for the ETL request to complete. This setting is useful for prototyping or for any time you want to be held in a wait state until the entire ETL Request has finished.

Starting ETL Requests With Your Own API Program

You can write a program that invokes an ETL Request from a remote platform. See the *iWay API Reference Manual* for details on your platform and desired calling language.

Example Writing a Command Sequence

In this example, the error logic performed after EDAACCEPT will depend on values returned by CMASAP to your program. There are two scb values that need to be reviewed:

- **scb.msg_origin.** While there may be multiple messages coming back from CMASAP (depending on the CM_SETERR value), you will always get the last message back with an origin of IVM. This is the critical message that your code should look for as it loops through the messages (check scb.msg_pending).
- **scb.msg_code.** A 18762 or 18763 indicates success. Anything else indicates a failure. Text of any failure can be found in scb.msg_text, and the most common possibilities are described in the following table:

Error	Description
18764	ETL Request has failed.
18528	ETL Request name was blank.
18529	ETL Request not found.
18561	ETL Request was not submitted.

Example C Program for Starting ETL Requests Remotely

The following is a sample C program for starting ETL Requests remotely.

Note: You must enhance this code as necessary to reflect production requirements.

```
// Run specified request

#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <ctype.h>

#define EDA_TYPES
#define EDA_VARIABLES
#define LINT_ARGS
#include "eda.h"
```

```
// chk_status - check status code in scb.  if non-zero
//          print out last command and status code
static long chk_status(EDA_SCB *scb_p,long status)
{
    char command[12];
    status = scb_p->status;  // add processing to decode status codes here
    if (status != 0)
        {strncpy(command, scb_p->command,12); command[12]='\0';
         printf("\n%s (%5ld) %s",command,status);
        }
    return(status);
}

void main()
{
    EDA_ID   *eid_p, eid;    // Pointer and EDA ID for this application.
    EDA_SCB  *scb_p, scb;    // Pointer and Session Control Block.
    long     status;         // Status return codes from API
    char     msg[3];         // EDA message origin

    char     server[]   = "SERVER";           // Replace with your server
    char     userid[]   = "USERID";           //                      userid
    char     password[] = "PASSWORD";         //                      password
    char     rpcname[]   = "CMASAP";           // Leave as CMASAP
    char     request[]   = "REQ_NAME=TESTING"; // Replace with request name

    long     s_len, p_len, u_len, c_len, r_len; // length of strings
    long     id, wait, zero = 0L;              // EDAAWAIT, TEST parms

    eid_p = &eid; scb_p = &scb;               // point to EDA structures

    // Get lengths of all the parms
    s_len = strlen(server);
    u_len = strlen(userid);
    p_len = strlen(password);
    c_len = strlen(rpcname);
    r_len = strlen(request);

    // Initialize EDA
    EDAINIT(eid_p, &status);
    if (status != 0) {printf("EDAINIT    (%5ld)",status); return;}

    // Connect to Server
    EDACONNECT(eid_p, scb_p, userid,&u_len, password,&p_len,
server,&s_len);
    if (chk_status(scb_p,status) != 0) {EDATERM(eid_p, &status); return;}

    // Invoke RPC
    EDARPC(scb_p, rpcname, &c_len, request, &r_len);
    if (chk_status(scb_p,status) != 0) {EDATERM(eid_p, &status); return;}
```

```

    EDATEST(scb_p,&wait);                // Eject Messages
    if (chk_status(scb_p,status) != 0) {EDATERM(eid_p, &status); return;}

    // Check for messages
    if (scb_p->msg_type)                  // Are there any messages?
    {if (scb_p->msg_code)                  // Is it really a message
        {strncpy(msg,scb_p->msg_org,3);    // First 3 chars (ie: EDA) are
clean
        msg[3]='\0';                      // Null terminate
        printf("\n%s (%4ld) ",msg, scb_p->msg_code);
        printf("%s", scb_p->msg_text);
    }
    while (scb_p->msg_pending)            // Any more messages?
    {EDAAcCEPT (scb_p);                 // Pull them out
        if (scb_p->msg_code)               // Are they really messages?
        {strncpy(msg,scb_p->msg_org,3); msg[3]='\0';
            printf("\n%s (%4ld) ",msg, scb_p->msg_code);
            printf("%s", scb_p->msg_text);
            if (scb_p->msg_code == 18762)    // See if we're done
                printf("Job submitted.\n");
        }
    }

    // Terminate connection
    EDATERM(eid_p, &status);
    return;
}

```

Starting ETL Requests From FOCUS

A FOCUS or WebFOCUS client on any platform is able to start ETL Requests. Notification of errors in execution will be returned to your application as error messages.

Syntax How to Start ETL Requests With a FOCUS or WebFOCUS Client

EX CMASAP parameters

where:

parameters

Are CMASAP parameters. For details, see *Starting ETL Requests With CMASAP* on page 5-13.

Determining the Status of an Event-Based ETL Request

When you schedule an event based execution, the ETL Request is scheduled for immediate execution. If resources are available, the ETL Request will receive its run-time parameters from the ETL Manager internal data store and begin its extraction and load as designed.

Submit the request with CMRUN. It sets a return code to the calling environment indicating if the request was submitted or completed successfully, depending on the setting for the Wait parameter.

You can also determine the status of an ETL Request after it has been submitted. This can be done in the following ways.

- Run the Copy Server Log (details) report. This allows you to view the Server Log file, which contains detailed information concerning the status of any given request. For details on the Copy Server Log (details) report see Chapter 9, *Generating Reports and Viewing the Log File*.
- Run the Statistics report and check the Server statistics.
- Check the iWay Web Console. To do this, select *iWay 51 ETL Manager Server* from the Programs menu in the Start menu while the request is running, and select *Agents* from the Monitor/Tune menu. For details on the iWay Workspace Web Console, see the installation and configuration guide for your platform.

CHAPTER 6

ETL Manager Tutorial

Topics:

- Before You Begin
- Building the ETL Request
- Scheduling and Running the ETL Request

This topic helps you create a sample ETL Request.

Before You Begin

The ETL Manager tutorial guides you through the creation of an ETL Request using the Workflow, Scheduler, and Procedures tools.

- The Workflow tool is used to build an ETL Request. For additional information on the Workflow tool and its components, see Chapter 4, *Building an ETL Request*.
- The Scheduler tool schedules the execution of your ETL Request. For additional information, see Chapter 5, *Scheduling and Executing ETL Requests*.
- The Procedures tool adds stored procedures to your ETL Request. For additional information, see Chapter 8, *Controlling ETL Requests With Procedures*.

Building the ETL Request


This topic describes how to build the ETL Request.

Selecting Data Sources

The first step in building the ETL Request is to select your data sources. Here you will:

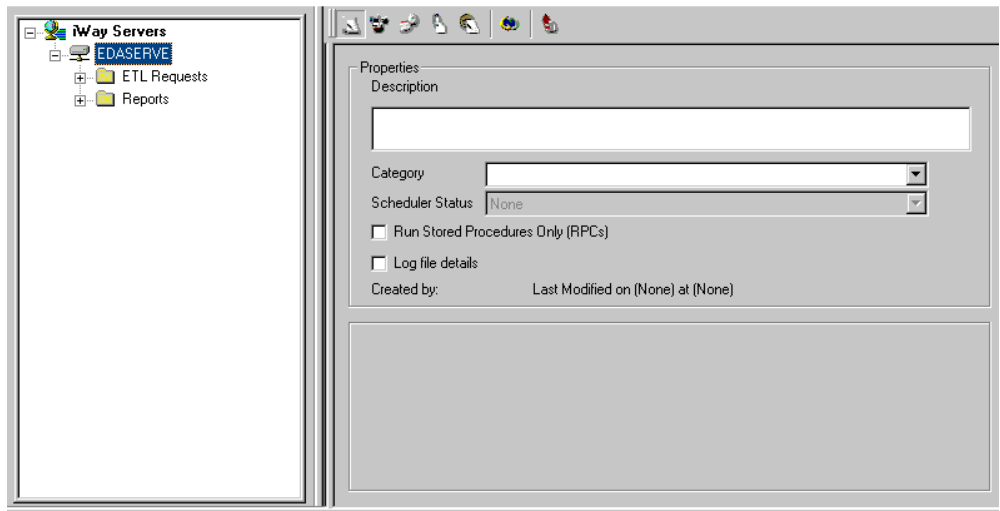
- Add both the CENTINV and CENTORD data sources to your request.
- Create a virtual column in the CENTORD data source.

Procedure How to Build the ETL Request

1. Do one of the following:
 - Select *New* from the File menu.
 - Right-click the *ETL Requests* folder, and select *New* from the pop-up menu.
 - Click the *New* button. 

If you are not connected to the selected server, the Enter Connection Information window opens so that you can connect. For details on connecting to a server, see *Connecting to the Server* on page 3-4.

The ETL Request opens.



2. In the Description field, enter *Build Fact Table*.
3. Click the *Workflow* tool.

The workspace and Components box opens. If necessary, move the Components box out of the way.

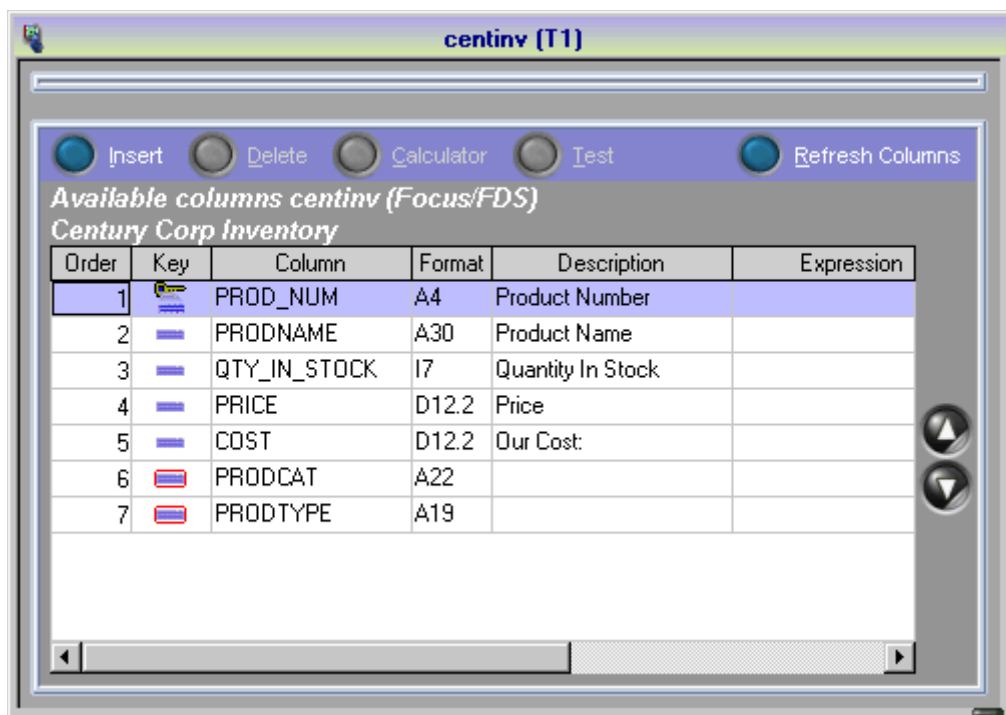


4. Add a Source component to the workspace.



5. Double-click the Source component to open it.
The Source window opens with a list of available data sources.
6. Scroll through the list of data sources until you find CENTINV, and double-click the gray box to the left of it to add the CENTINV data source to your request.

A list of the columns in the CENTINV data source opens, and the list of sources is hidden.



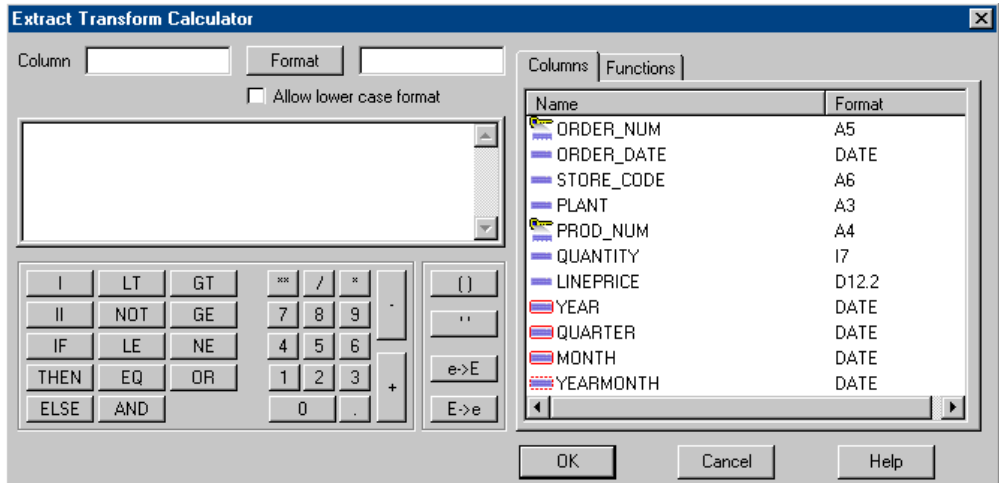
7. Double-click the control icon on the title bar to close the Source window.
8. Add another Source component to the workspace and drop it next to the first Source component.



9. Double-click the new Source component to open it.
The Source window opens.
10. Scroll through the list of data sources until you find CENTORD, and double-click the gray box to the left of it to add the CENTORD data source to your request.
A list of the columns in the CENTORD data source opens, and the list of sources is hidden.
11. Create an extract transform by clicking *Insert*.

12. Click *Calculator*.

The Extract Transform Calculator opens.

**13. Enter *YEARMONTH* in the Column field.****14. Enter *DATE* in the Format field.****15. Create your calculation:**

- a.** Double-click *YEAR* in the Columns window.
- b.** Enter ** 100 +* in the Expressions window.
- c.** Double-click *MONTH* in the Columns window.

16. Click *OK*.

The *YEARMONTH* column appears in the Columns table.

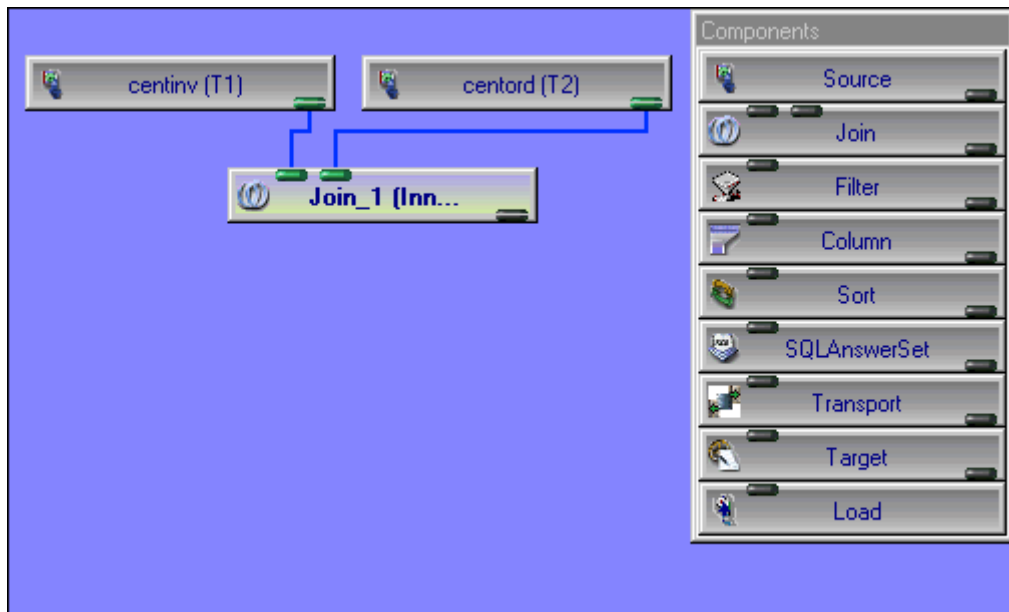
17. Double-click the control icon on the title bar to close the Source component.

Joining Data Sources

Since you selected two data sources, you must join them. You will join the CENTINV and CENTORD data sources in an inner join using the PROD_NUM fields.

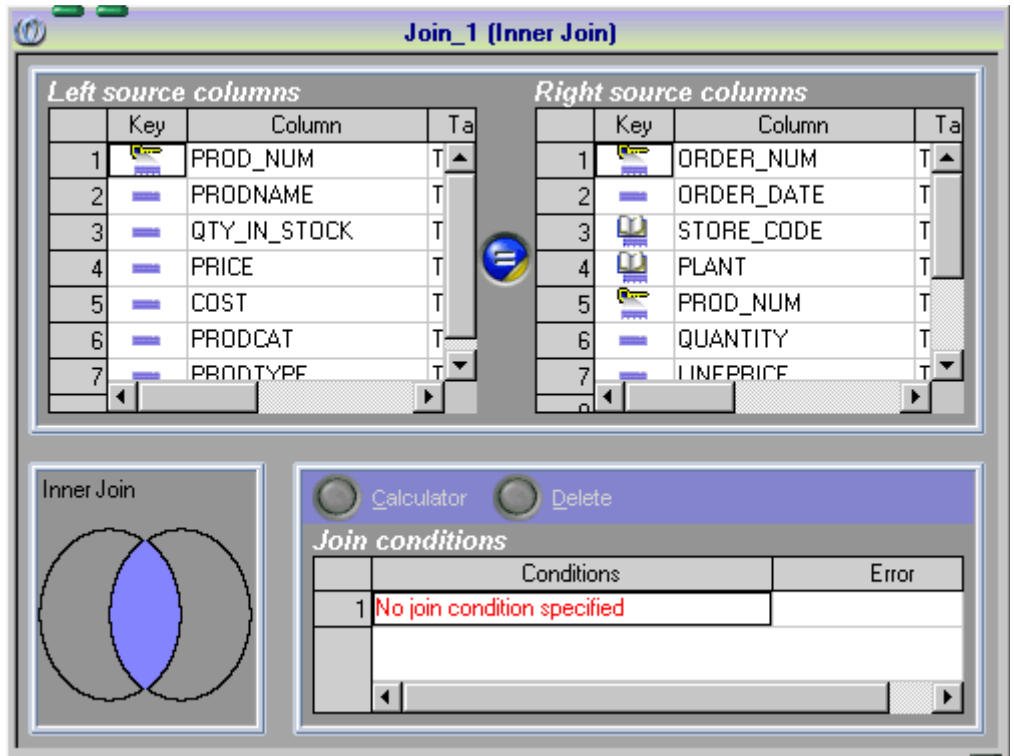
Procedure How to Join Data Sources

1. Add a Join component to the workspace under the Source components, and connect it to both source components.



2. Double-click the Join component to open it.

The Join window opens.



3. Join the CENTINV and CENTORD data sources with the PROD_NUM fields:
 - a. In the Left source columns table, select *PROD_NUM*.
 - b. In the Right source columns table, select *PROD_NUM*.
 - c. Click .

Do not click the Join Definition diagram. The default is an inner join. The join condition appears in the Join Conditions window.

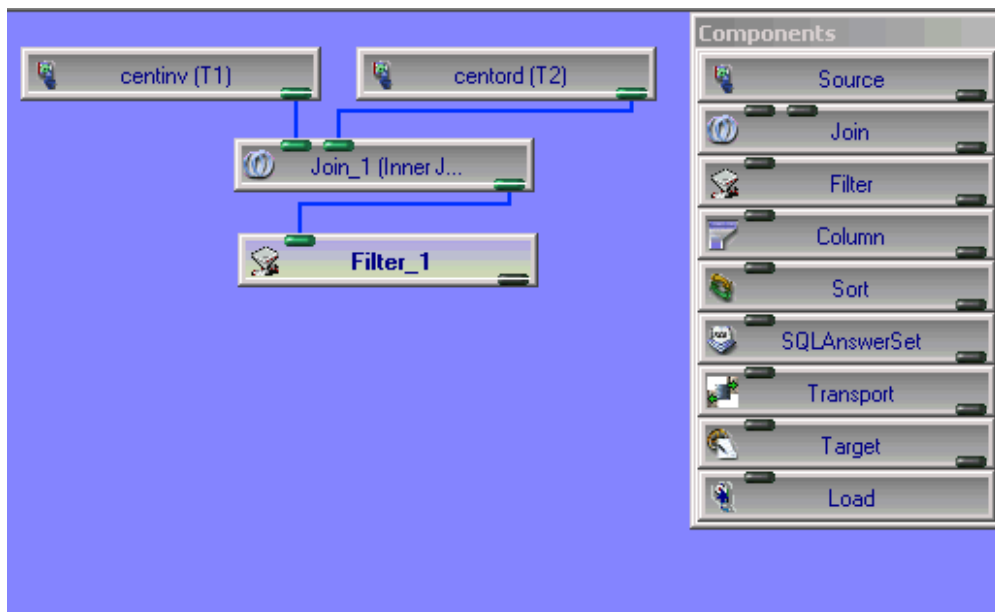
4. Double-click the control icon on the title bar to close the Join component.

Adding Filters to the ETL Request

You can add filters to your copy request. Here you will select only data with the year 2000.

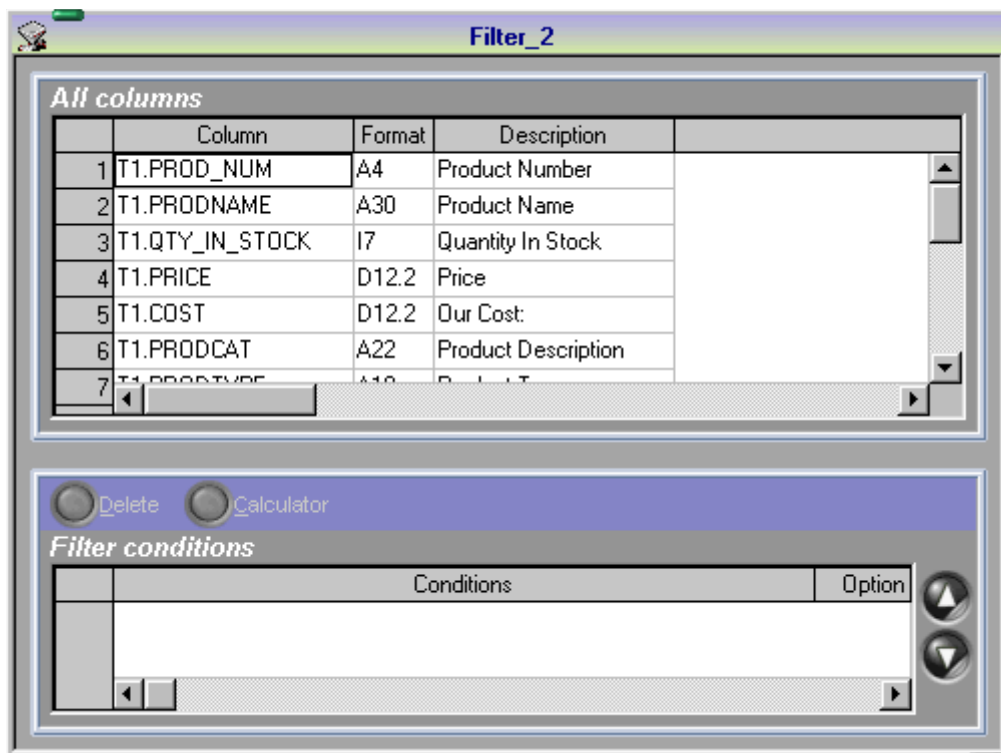
Procedure How to Add Filters

1. Add a Filter component to the workspace, and connect the Filter component to the Join component.



2. Double-click the Filter component to open it.

The Filter window opens.



3. Double-click *YEAR* in the All columns table.
The *YEAR* column appears in the Filter conditions table.
4. Double-click in the Conditions field and add =2000.
5. Double-click the control icon on the title bar to close the Filter window.

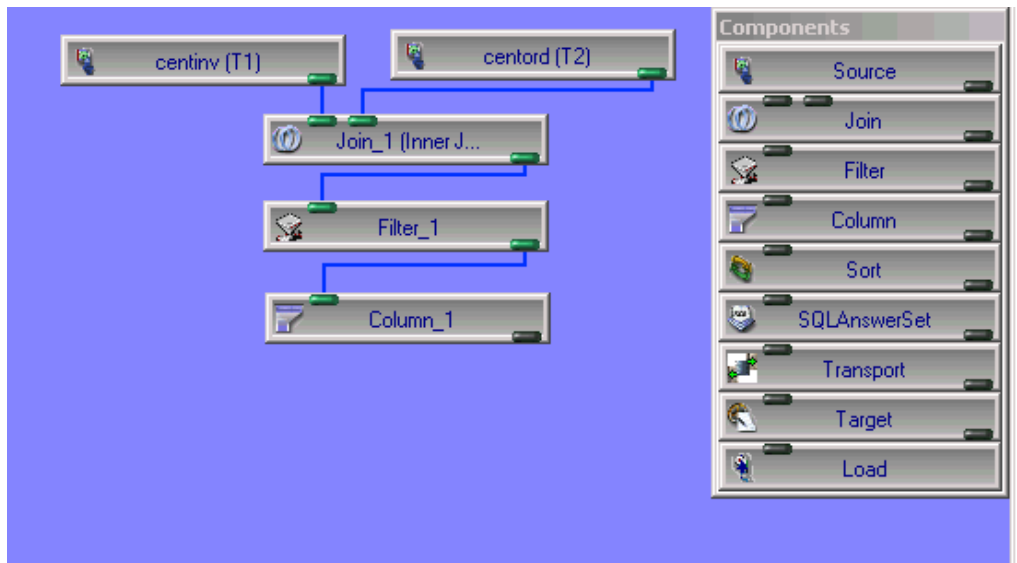
Selecting Columns

You can now select the columns needed for the facts table. You will do the following:

- Add the product number (PROD_NUM) and store code (STORE_CODE) columns to the ETL Request.
- Designate the time period in which each record will be stored by adding the YEARMONTH field.
- Add the columns for quantity (QUANTITY) and line price (LINEPRICE) so you can measure sales.
- Calculate the line item cost by adding a new column (LINECOGS) which includes an SQL calculation.
- Group your key columns since we are creating a month level detail.
- Sum the quantity, line price, and new column.

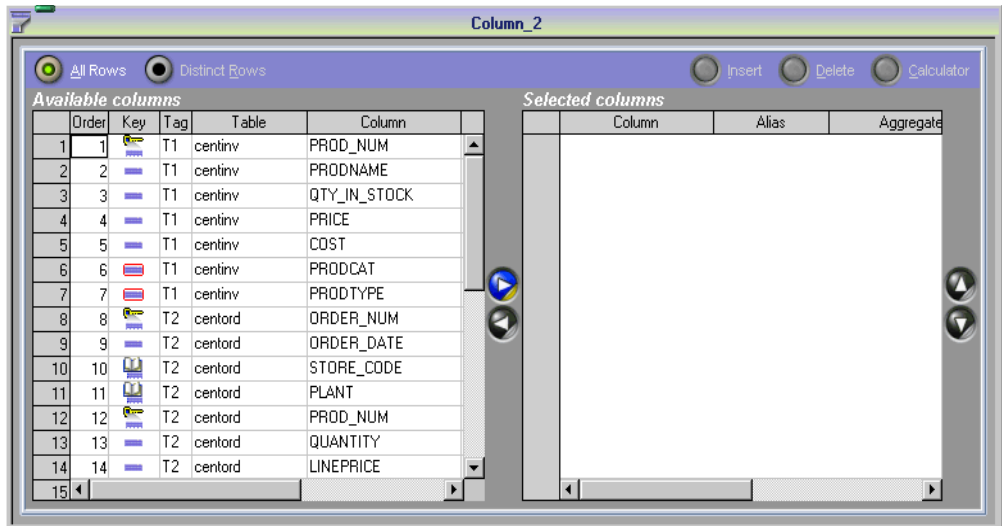
Procedure How to Select Columns for the ETL Request

1. Add a Column component to the workspace under the Filter component, and connect the Column component to the Filter component.



2. Double-click the Column component to open it.

The Column window opens.



3. Scroll down the list of columns in the Available columns table, and select the following columns by double-clicking the gray boxes to the left of them:

- T2.PROD_NUM
- T2.STORE_CODE
- T2.YEARMONTH
- T2.QUANTITY
- T2.LINEPRICE

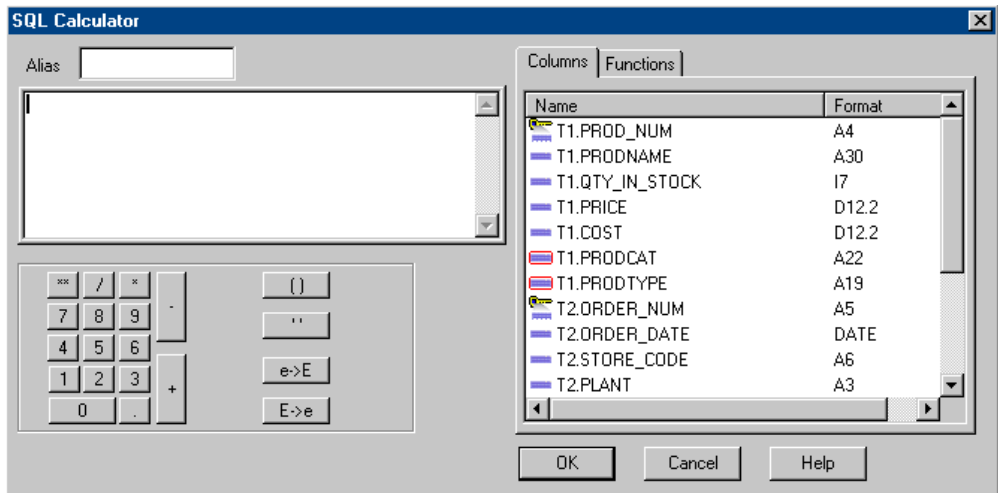
The columns display in the Selected columns table.

4. Click *Insert* to create a new column.

A new column is added to the Selected columns table.

5. Click *Calculator*.

The SQL Calculator opens.



6. Enter *LINECOGS* in the Alias field.

7. Create your calculation for the new field:

- Double-click *T2.QUANTITY*.
- Click ***.
- Double-click *T1.COST*.

8. Click *OK*.

The Column window is active again, and *T2.QUANTITY*T1.COST* appears in the Column field for the new column.

9. Aggregate the key columns:

- Select *PROD_NUM*, *STORE_CODE*, and *YEARMONTH* from the Selected columns table.
- Double-click the Aggregate field and select *GroupBy* from the drop-down list.

10. Sum the *QUANTITY*, *LINEPRICE*, and *LINECOGS* columns:

- Select *QUANTITY*, *LINEPRICE*, and *LINECOGS* from the Selected columns table.
- Double-click the Aggregate field and select *Sum* from the drop-down menu.

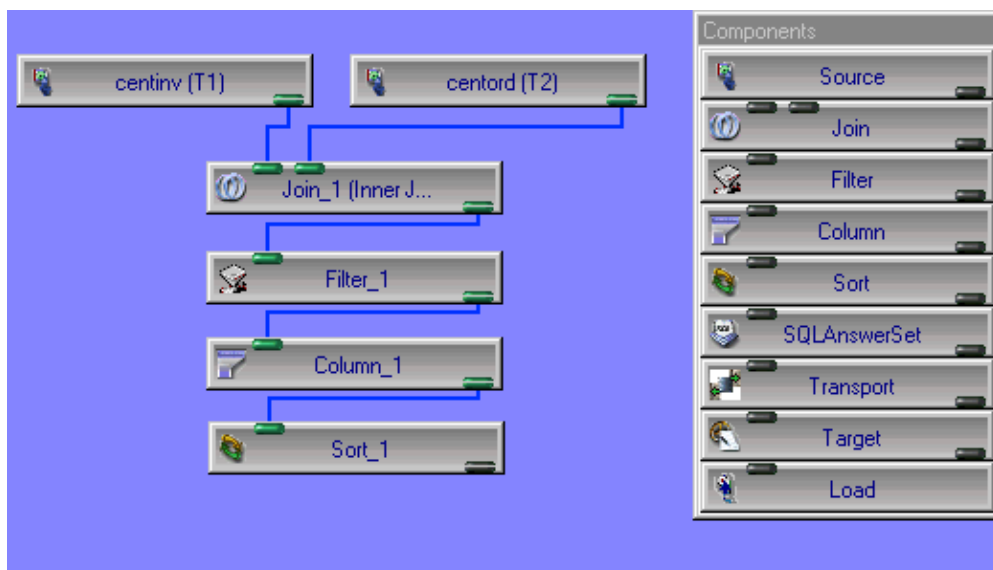
11. Double-click the control icon on the tile bar to close the Column component.

Specifying a Sort

You can determine the order in which new columns appear in the results by sorting the data.

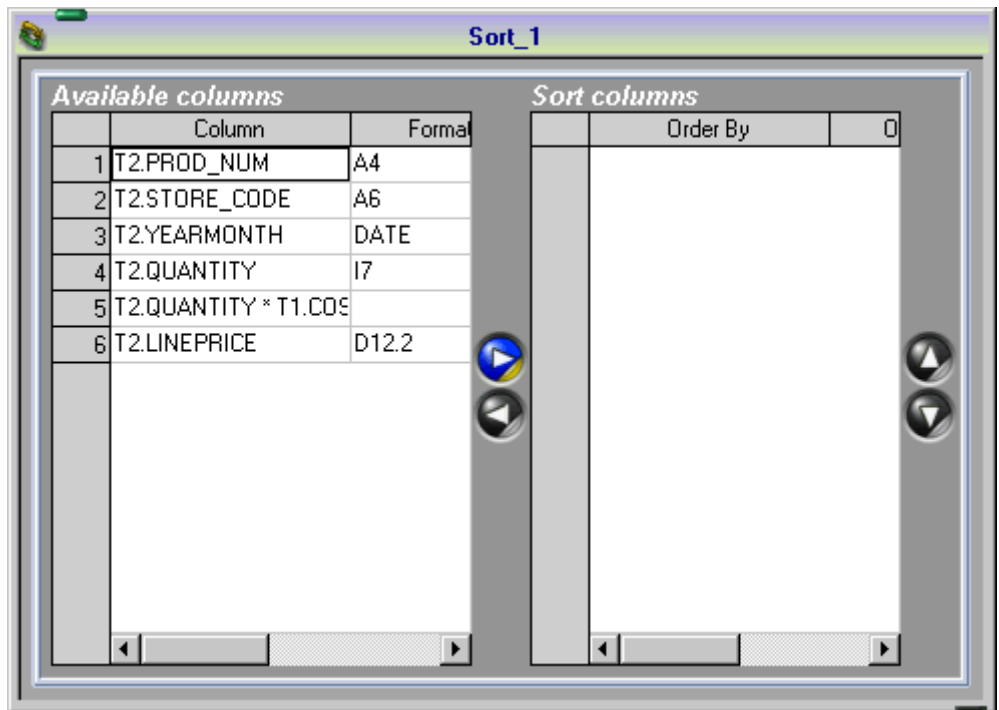
Procedure How to Specify a Sort

1. Add a Sort component to the workspace under the Column component, and connect the Sort component to the Column component.



2. Double-click the Sort component to open it.

The Sort window opens.



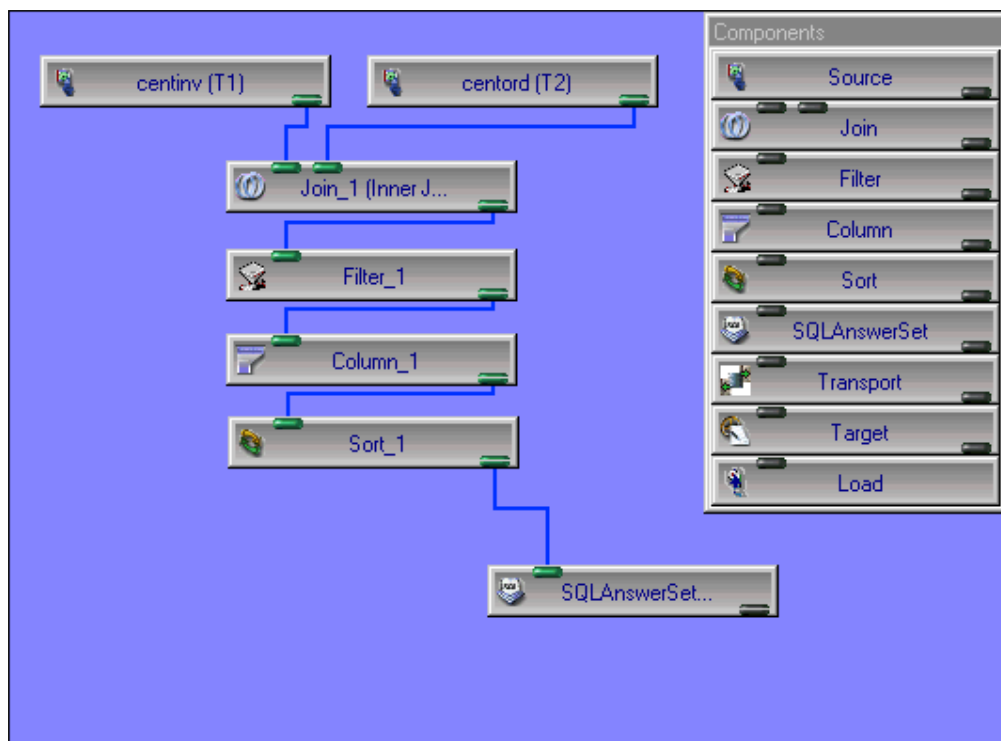
3. Select the *STORE_CODE* column by double-clicking the gray box to the left of it. *STORE_CODE* appears in the Sort columns grid.
4. Double-click the control icon on the tile bar to close the Sort component.

Testing the SQL

You can test the SQL created by your ETL Request.

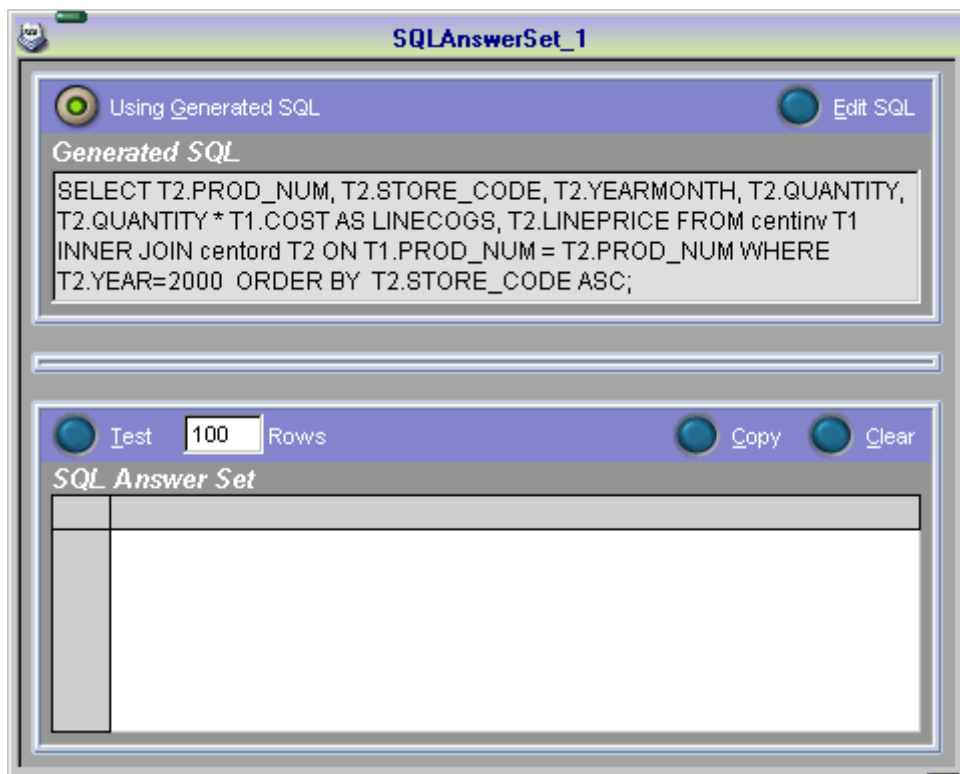
Procedure How to Test the SQL

1. Add an SQLAnswerSet component to the workspace, and connect it to the Sort component.



2. Double-click the SQLAnswerSet component to open it.

The SQLAnswerSet window opens.



3. Click *Test*.

The results of the current SQL appear in the bottom window.

4. Double-click the control icon on the title bar to close the SQLAnswerSet window.
5. Right-click the SQLAnswerSet component, and select *Delete* from the pop-up menu.

Identifying the Target Table and Specifying Mappings

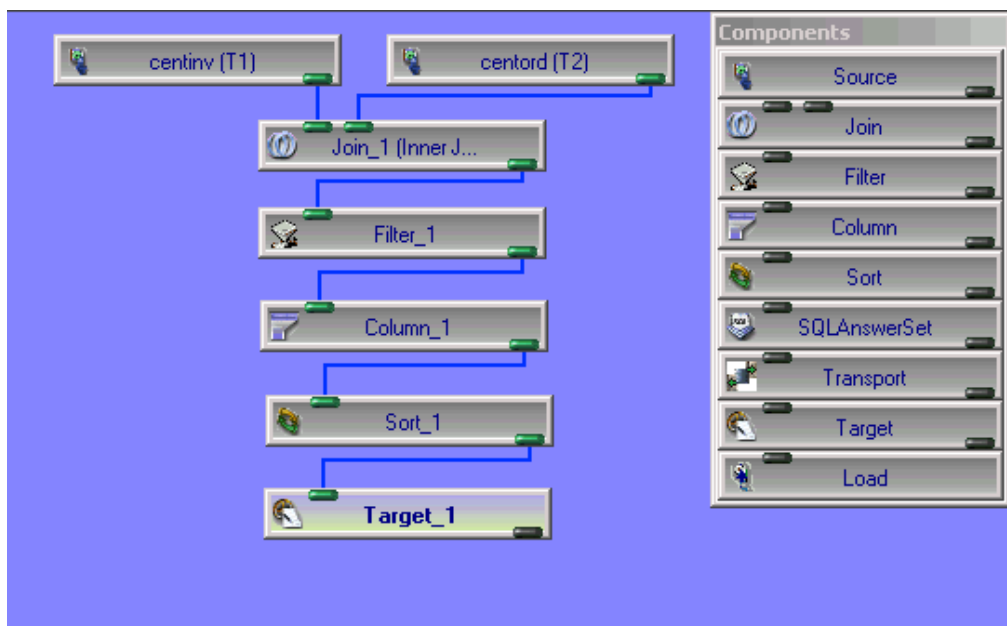
You can use either a new or existing table as the data target for your request, and can then specify mappings for your ETL Request. Here you will do the following:

- Select the existing CENTFACT table as your data target.
- Automap the columns that have matching columns from both tables.
- Use simple mapping to map YRMTH to YEARMONTH.
- Create a complex mapping for the PROFIT field.
- Create a transformation that validates records.
- Test your mappings.

Note: You can specify several options in the Target component that you do not utilize in this procedure. For information on these options see Chapter 4, *Building an ETL Request*.

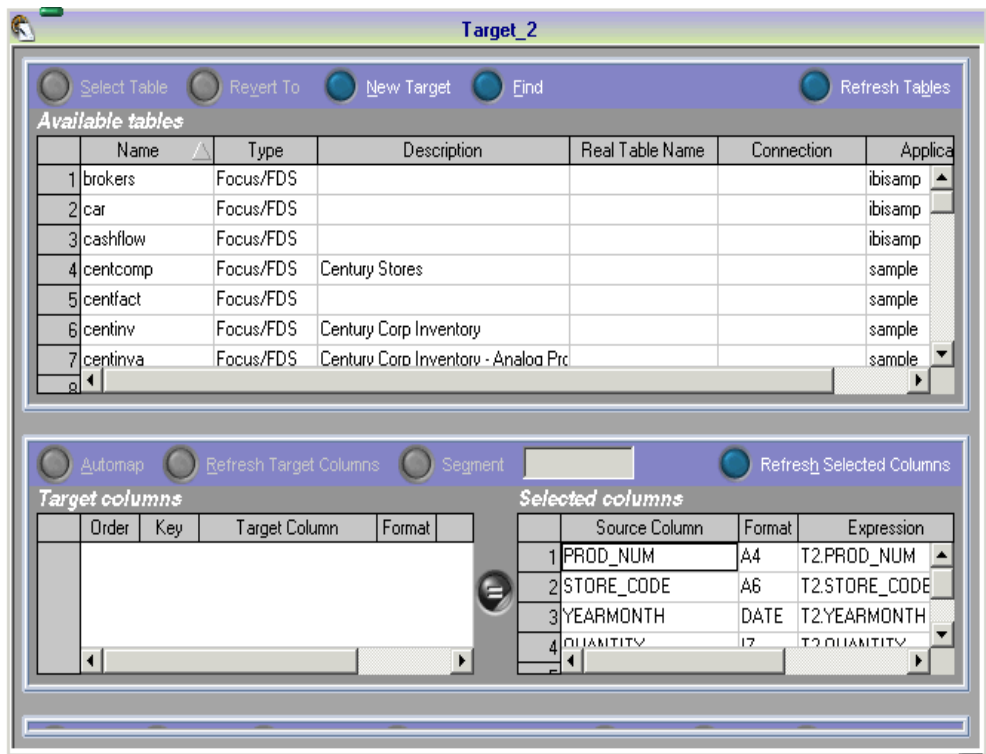
Procedure How to Identify the Target Table and Mappings

1. Add a Target component to the workspace, and connect it to the Sort component.



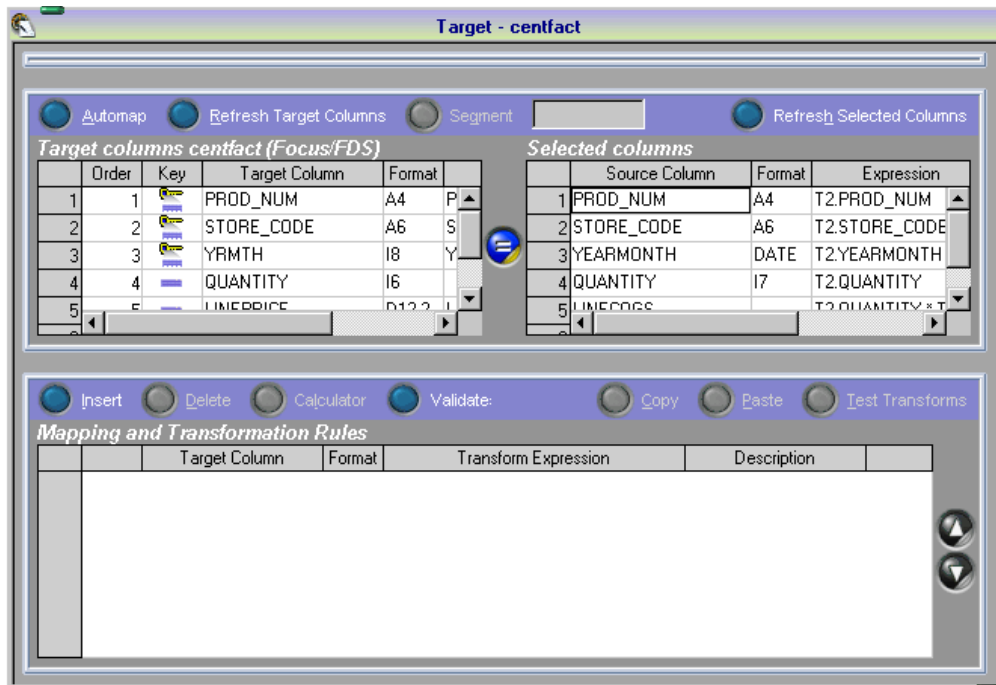
2. Double-click the Target component to open it.

The Target window opens.



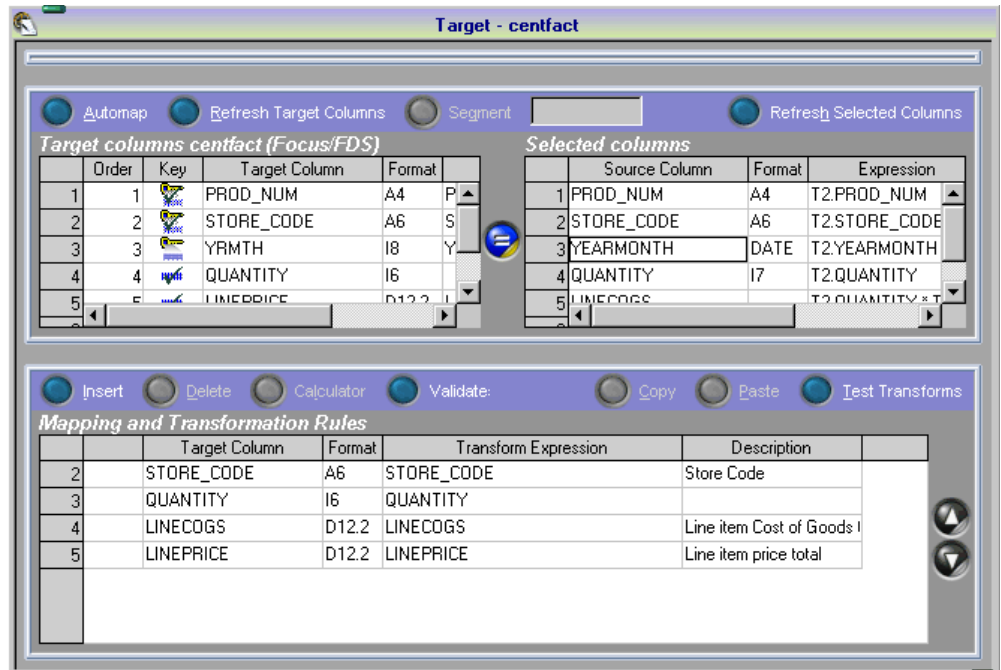
3. Double-click the *CENTFACT* table.


The *CENTFACT* columns display in the Target columns table, and the Mapping and Transformation Rules table opens below.



4. Click *Automap*.

All columns with identical names are mapped, and they display in the Mapping and Transformation Rules window. These columns are also checked off in the Target columns table. Since YRMTH and PROFIT do not have matching columns, they are not mapped.

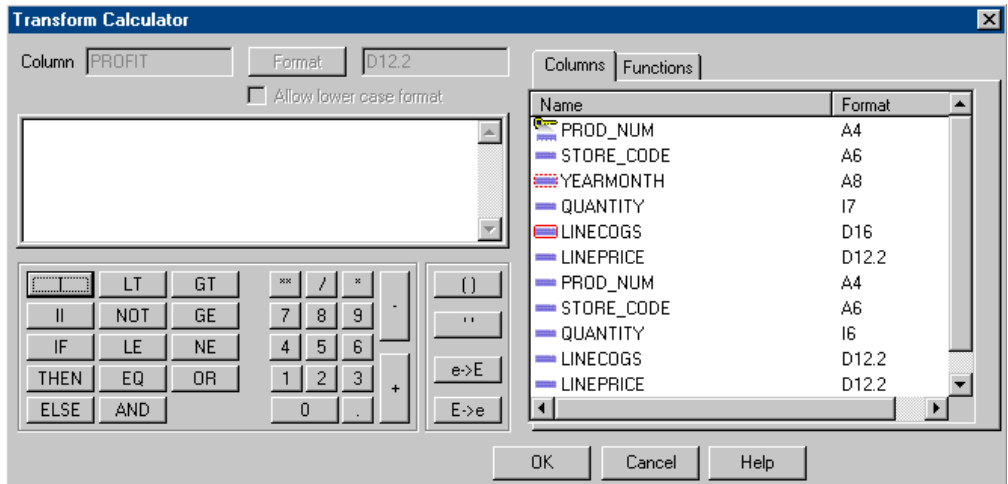


5. Select *YRMTH* from the Target columns table, select *YEARMONTH* from the Selected columns table, and click .

The mapping of YRMTH to YEARMONTH appears in the Mapping and Transformation Rules table.

6. Double-click *PROFIT* to move it to the Mapping and Transformation Rules table.
7. Select *PROFIT* from the Mapping and Transformation Rules table, and click *Calculator*.

The Transform Calculator opens:



8. Create your mapping for the PROFIT column:
 - a. Double-click *LINEPRICE* in the Columns window.
 - b. Click *-*.
 - c. Double-click *LINECOGS* in the Columns window.
9. Click *OK*.
10. Click *Validate*.

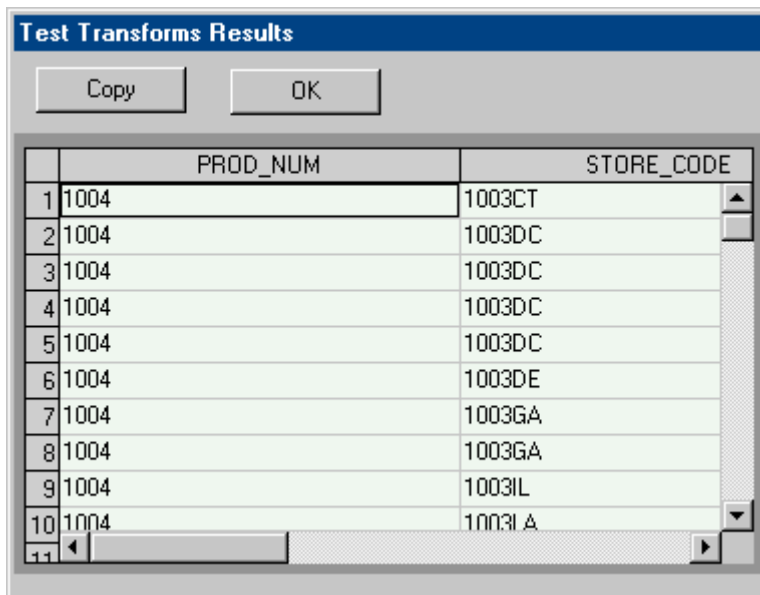
A *VALIDATE* row is created.
11. Click *Calculator*.

The Transform Calculator opens.
12. Enter *TEST1* in the Column field.
13. Enter *I6* in the Format field.
14. Create your validate criteria:
 - a. Double-click *QUANTITY* in the Columns window.
 - b. Click *GT*.
 - c. Click *0*.
15. Click *OK*.

You return to the Target (Mapping) window.

16. Click *Test Transforms* to test your mappings.

The answer set appears in the Test Transforms Result window:



The screenshot shows a window titled "Test Transforms Results". At the top, there are two buttons: "Copy" and "OK". Below the buttons is a table with two columns: "PROD_NUM" and "STORE_CODE". The table contains 10 rows of data. The first column has a small index from 1 to 10. The second column has a small index from 1 to 10. The table is scrollable, as indicated by the scrollbar on the right.

	PROD_NUM	STORE_CODE
1	1004	1003CT
2	1004	1003DC
3	1004	1003DC
4	1004	1003DC
5	1004	1003DC
6	1004	1003DE
7	1004	1003GA
8	1004	1003GA
9	1004	1003IL
10	1004	1003LA

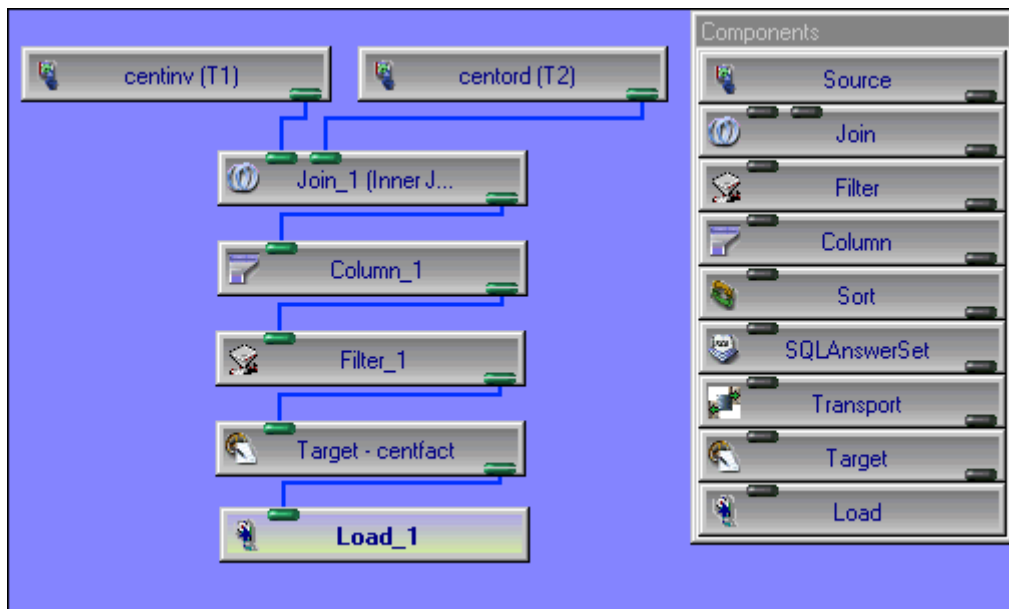
17. Click OK.

Specifying Load Options

You can now specify load options.

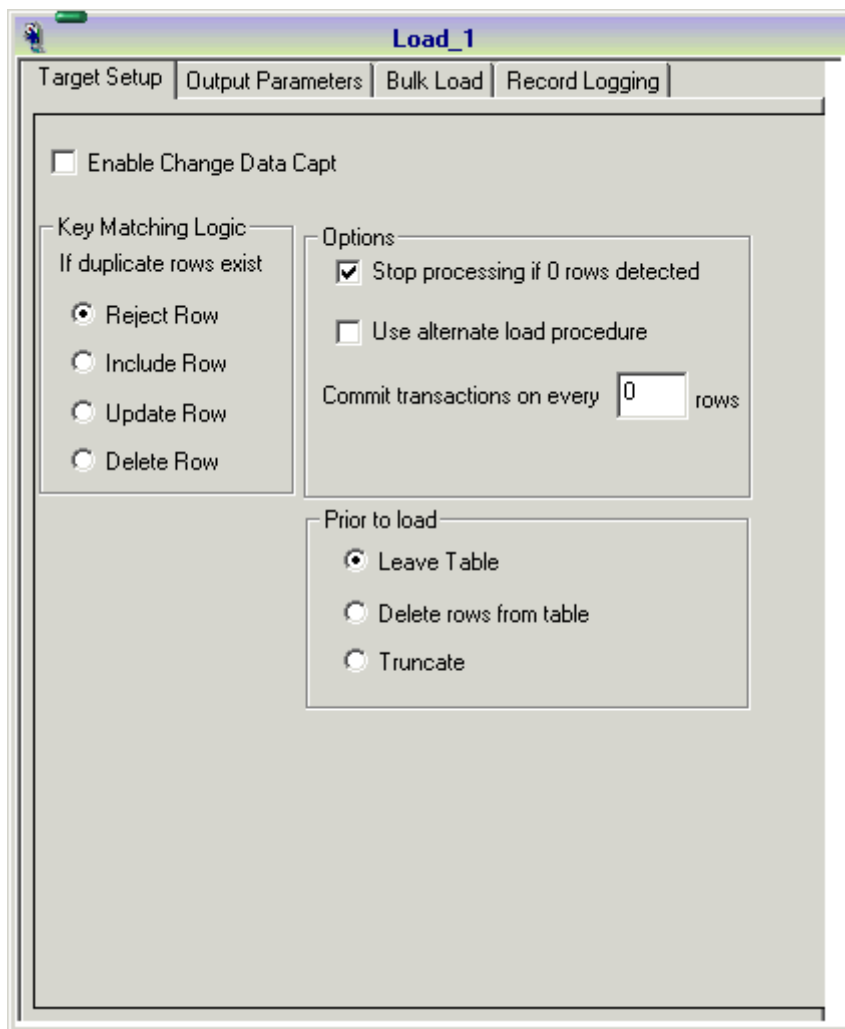
Procedure How to Specify Load Options

1. Add a Load component to the workspace, and connect the Load component to the Target component.



2. Double-click the Load component to open it.

The Load window opens.



3. Ensure that *Reject Row* is selected in the Key Magic Logic If duplicate row exists section, and *Stop processing if 0 rows detected* is selected in the Options section.
4. Double-click the control icon on the title bar to close the Load window.

Scheduling and Running the ETL Request

This topic describes how to schedule and run an ETL Request using the Scheduler tool. You can also view the log of your ETL Request.

Scheduling the ETL Request

You can schedule your ETL Request to run once or several times using the Scheduler tool. Here you will set your ETL Request to run on a recurring schedule.

Procedure How to Schedule the ETL Request

1. Click the *Scheduler* tool.

The Scheduler opens:

The Scheduler window is divided into several sections:

- Schedule Type:**
 - ☒ Run Once
 - ☐ Recurring
 - ☐ Multi-Days
- Start Time:**
 - HH: 12, MM: 49, DD: 10, MM: 09, YYYY: 2001
- End Time:**
 - HH: 00, MM: 00, DD: , MM: , YYYY:
- Interval Type:**
 - ☒ Hourly
 - ☐ Daily
 - ☐ Weekly
 - ☐ Monthly
 - ☐ Yearly
- Interval Number:**
 - 00
- Days of the Week:**

Sun	Mon	Tue	Wed	Thu	Fri	Sat
-----	-----	-----	-----	-----	-----	-----
- Days of the Month:**

1	2	3	4	5	6	7
8	9	10	11	12	13	14
15	16	17	18	19	20	21
22	23	24	25	26	27	28
29	30	31	<input type="checkbox"/> Last Day of Month			

2. In the Schedule Type section, click *Recurring*.
The areas of the Scheduler window that were grayed out now become available.
3. In the Interval Type section, click *Daily*.

4. In the Start Time section:
 - a. Enter 8 in the HH (hour) field.
 - b. Enter 00 in the MM (minutes field).
 - c. Leave the date that is displayed. The schedule will start at the current date.

Running the ETL Request

You can now run your ETL Request.

***Procedure* How to Run the ETL Request**

1. From the Actions menu, select *Run*.

Since you have not saved the request, ETL Manager asks if you would like to save the Request before running it.
2. Click *Yes*.

You are asked if you want to verify the query.
3. Click *No*.

The Save As... dialog box opens.
4. Enter *DEMO* in the File name field, and click *OK*.

The Uploading window opens, and the ETL Request is saved. The Immediate Execution window then opens.
5. Click *OK*.
6. The ETL Request runs.

Viewing the ETL Request Log

You can view a record of the copy processes and any problems with the logs.

***Procedure* How to View the ETL Request Log**

1. Right-click your request in the ETL Request folder.
2. Select *View Last Log* from the pop-up menu.

The results are retrieved and displayed.

CHAPTER 7

Working With an ETL Request

Topics:

- Editing an ETL Request
- Saving an ETL Request
- Printing an ETL Request
- Copying an ETL Request
- Deleting an ETL Request
- Exporting an ETL Request
- Executing an ETL Request Immediately
- Specifying ETL Request Properties
- Filtering the Display of ETL Requests

This topic provides instructions on managing an ETL Request. For information on creating an ETL Request, see Chapter 4, *Building an ETL Request*.

Editing an ETL Request


You can edit an existing ETL Request in this folder with the same tools you used to create, schedule, and customize the ETL Request.

Procedure How to Edit an ETL Request


1. Double-click the ETL Request to open it.

The ETL Request main window opens.


2. Select the tool you need for the type of change you want to make.

- To edit the source, target, joins, mappings, filters, or sorting, select the *Workflow* tool. 


For details on using the Workflow tool, see Chapter 4, *Building an ETL Request*.

- To change the execution schedule, select the *Scheduler* tool. 

For details on using the Scheduler tool, see Chapter 5, *Scheduling and Executing ETL Requests*.

- To change dependencies, select the *Dependencies* tool. 

For details on using the Dependencies tool, see Chapter 5, *Scheduling and Executing ETL Requests*.

- To add or delete procedures, select the *Procedures* tool. 

For details on using the Procedures tool, see Chapter 8, *Controlling ETL Requests With Procedures*.

3. Make your changes.

Saving an ETL Request

When you make a change to an ETL Request, you must save your request for the changes to take effect.


If changes have been made to the ETL Request that have not been saved, the *Save* button is available and the ETL Request title bar will display an asterisk. If changes have not been made, the *Save* button is not available, and an asterisk does not display.

The following problems may occur when saving an ETL Request:

- If another user has saved an ETL Request with the same name, a dialog box is displayed to advise you of this and lets you save the request with a different name.
- If ETL Manager is unable to save the ETL Request, a message appears in the message window. This may occur if the disk used to store ETL Requests is full, or if you do not have permission to write to the specified directory.

Procedure How to Save an ETL Request

1. Do one of the following:

- Click the *Save* button. 
- or
- Select *Save* from the File menu.

The Save dialog box opens.

2. Enter a name for the request in the File name field, and click *Save*. The request's name must be in lowercase letters. If the name is entered in mixed case, ETL Manager converts the name to lowercase.

The Uploading window opens, and ETL Manager saves your request.

When the ETL Request is successfully saved, the following message appears:

(ICM18164) Request request created/updated successfully by user: user

If the ETL Request is not saved successfully, the following message appears:

(ICM1862) Error saving request request user user

Note: You can control the application in which an ETL Request is saved by entering an application name in the Category field of the Properties window.

Printing an ETL Request

You can print an ETL Request if it is open and selected.

Procedure How to Print an ETL Request

1. Open the ETL Request.
2. Click the *Print* icon, or select *Print* from the File menu.

Note: You can display the image on screen before printing to the printer by selecting *Print Preview* from the File menu.

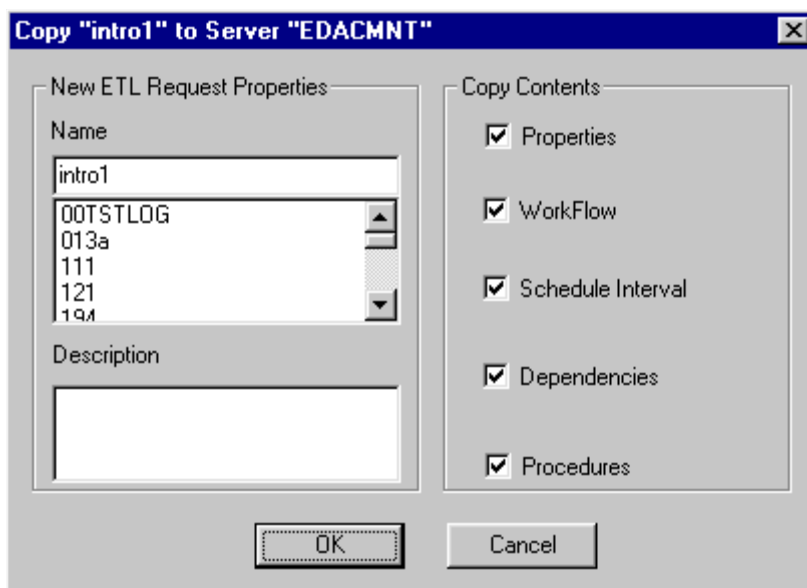
Copying an ETL Request

You can copy an ETL Request from one server to another. Copying an ETL Request copies data into a flat file on a target server and then loads it into the target server internal ETL Manager data store.

Procedure How to Copy an ETL Request

1. Highlight or open the ETL Request.
2. Select *Copy* from the Edit menu.
3. Select *Paste* from the Edit menu.

The Copy to Server window opens:



4. Select the information in the ETL Request you want to copy:
Properties is the properties information specified in the ETL Request main window.
Workflow is the information about sources, targets, joins, mappings, sorting, and filters.
Schedule Interval is the execution schedule for the ETL Request.
Dependencies are the ETL Requests scheduled to execute after the successful completion of another ETL Request.
Procedures are the stored procedures scheduled to execute with the ETL Request.
5. Enter a name for the ETL Request in the Name field, or select one from the displayed list.
6. Enter a description for the ETL Request in the Description field.
7. Click OK.

Procedure How to Move an ETL Request to Another Server Using Drag and Drop

1. Log on to both the server that contains the ETL Request and the server to which you will copy the ETL Request.
2. Select the ETL Request you want to copy.
3. Drag the ETL Request to the server you want to copy it to.

The ETL Request is copied to the selected server.

Deleting an ETL Request

You can delete a saved ETL Request. The ETL Administrator ID can delete any ETL Request.

Procedure How to Delete an ETL Request

1. Select an ETL Request from the ETL Requests folder.
2. Do one of the following:
 - Select *Delete* from the Edit menu.
 - Right-click the request, and select *Delete* from the pop-up menu.
 - Press the Delete key.

A prompt appears, verifying that you want to delete the ETL Request.

3. Click *OK* to continue, or *Cancel* to cancel the deletion.

A message appears in the message log that indicates whether the request was deleted or not.

Example **Deletion Messages**

If the ETL Request was deleted, a message similar to the following appears:

(ICM18171) Complete deleting: request for User: user

If the request could not be deleted, a message similar to the following appears:

(ICM18193) Error deleting request, User user

Failing to Delete an ETL Request

An ETL Request deletion may fail. This may occur if the ETL Request has already been deleted, the request belongs to someone else, or the ETL Request does not exist.

Exporting an ETL Request

You can export an ETL Request from ETL Manager as an ASCII text file or as XML.

- Exporting as an ASCII text file allows you to e-mail the ETL Request to someone.
- Exporting an ETL Request as an XML file allows you to supply the ETL Request to Information Builders' customer support when reporting a problem.

Procedure **How to Export an ETL Request as Text**

1. Open the ETL Request.
2. Select *Export* from the Actions menu, or click the *Export* button in the ETL Request toolbar.
The Export dialog box opens.
3. Select *Clipboard* or *File* to designate where to export the ETL Request to.
 - If you select *Clipboard*, the ETL Request is saved to the clipboard.
 - If you select *File*, the Save As dialog box opens. Select a location and file name, and click *Save*.

Procedure **How to Export an ETL Request as XML**

1. Open the ETL Request.
2. Select *Export to XML* from the Actions menu.
The Diagnostic: Export Request to XML dialog box opens.
3. Select a location and file name, and click *Save*.

Executing an ETL Request Immediately

An ETL Request can be executed immediately by running it while the request is open or selected.

You can also execute an ETL Request at a scheduled execution time or dependent upon the execution of another request. For more information on scheduling and executing requests, see Chapter 5, *Scheduling and Executing ETL Requests*.

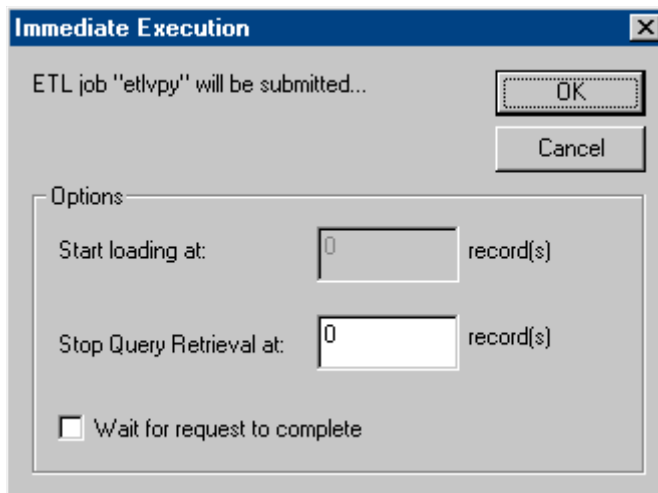
Procedure How to Execute an ETL Request Immediately

1. Select an ETL Request from the ETL Requests folder.
2. Do one of the following:
 - Right-click the ETL Request and select *Run* from the context menu
 - Open the ETL Request and click the *Run* icon in the ETL Request toolbar.
 - Open the ETL Request and select *Run* from the Actions menu.

The Immediate Execution dialog box opens. For details on the Immediate Execution dialog box see *Immediate Execution Dialog Box* on page 7-7.

3. Enter values in the *Start loading at:* and *Stop Query Retrieval at:* fields and make sure *Wait for request to complete* is not selected.
4. Click *OK*.

Reference Immediate Execution Dialog Box



The Immediate Execution dialog box contains the following fields/options:

Start Loading at:

Is the record number to start loading the data target at. For example, if you enter 20, the first 19 records would be skipped, and record 20 would be the first record loaded.

This option is not available when the target is a flat file or server transfer file.

Stop Query Retrieval at:

Indicates the record number at which to stop retrieving records.

If this option is used, the request will not use automatic pass-through. For more information on automatic pass-through see Chapter 13, *Improving Performance*.

Wait for the request to complete

If this option is unchecked, the request is submitted and control is returned to you so that you can continue to work with ETL Manager Workbench. If checked, control is returned when the request is complete.

Note: In some cases you may want to wait for the ETL Request to complete, especially in testing and development or when the request will take just a few minutes. If you check this box, the wait cursor (usually an hourglass) will remain on the screen until the request completes. You can then view the log. This is useful during initial development of your ETL Requests.

Specifying ETL Request Properties

You can specify ETL Request properties such as a description, category, and schedule status in the ETL Request main window. You can specify the following:

- A text description that describes the ETL Request. It appears in the ETL Requests folder beside the request name.
- An application which the ETL Request is part of. This determines the directory in which the ETL Request is saved. An ETL Request can only be saved to the directories specified in the current APP PATH.
- A schedule status that specifies whether the ETL Request is scheduled and will execute.
- A checkbox to indicate that ETL processing should be bypassed and only user-specified procedures should be run.

Procedure How to Specify ETL Request Properties

1. Open the ETL Request.

The ETL Request main window opens.

2. Enter a description for the ETL Request in the Description field.
3. Select a category from the Category drop-down list to categorize your ETL Request.
4. Select a status from the Scheduler status drop-down list:

Active indicates the ETL Request is complete and the Scheduler will execute it.

Inactive indicates the ETL Request is complete, but the Scheduler will not execute it. You can still execute inactive ETL Requests with the Run button.

None indicates the ETL Request is not scheduled.

5. Select *Run Stored Procedures Only (RPCs)* if you want to run the procedures associated with the ETL Request, but not execute the request.

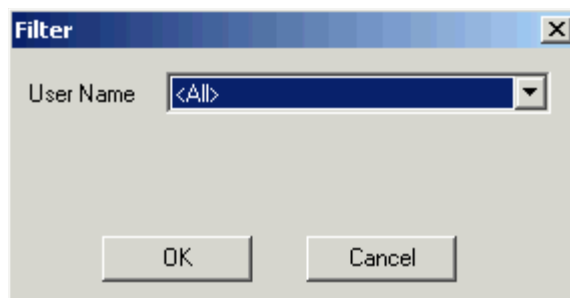
Filtering the Display of ETL Requests

You can limit the ETL Requests displayed in your ETL Requests folder by filtering the ETL Requests. You can filter by user name. This displays only the ETL Requests saved by the selected user name.

Procedure How to Filter the Display of ETL Requests

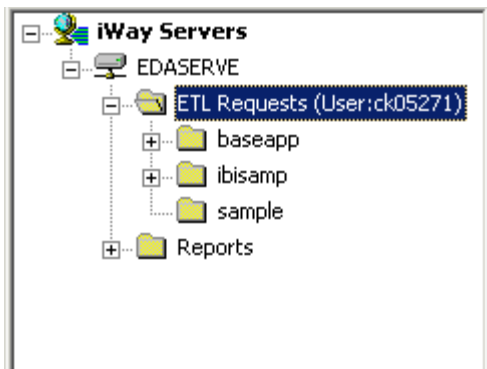
1. Right-click the ETL Requests folder, and select *Filter* from the pop-up menu.

The Filter dialog box opens.



2. Select a user name from the User Name drop-down list to display only the ETL Requests saved by that user.
3. Click OK.

The list of ETL Requests appears for the user you selected.



CHAPTER 8

Controlling ETL Requests With Procedures

Topics:

- Executing a Procedure With an ETL Request
- Executing a Procedure Without an ETL Request
- Running a Stored Procedure
- Changing Run-Time Behavior With CMOPTION

This topic describes how to use procedures to perform tasks that are not automated by ETL Manager. For additional information on procedures, see the Server documentation for your platform and the *iWay Stored Procedures Reference Manual*.

Executing a Procedure With an ETL Request

Although ETL Manager automates as many tasks as possible, you may need to perform some specialized tasks manually while executing an ETL Request. These tasks may include sending e-mail at the start and completion of a job, setting up the environment, and copying files. ETL Manager allows you to perform these tasks by executing procedures before, during, and after executing the ETL Request. These procedures can be any procedure that iWay can execute.

Procedures can be executed with the Procedures tool. ETL Manager includes some pre-defined procedures and variables that enable you to change ETL Manager's behavior without changing settings at the Server.

You can execute the following types of procedures:

- **Pre-Extract** procedures run before the ETL Request extracts data from the source.
- **Post-Extract** procedures run after the ETL Request extracts data from the source.
- **Pre-Load** procedures run before the ETL Request loads data into the data target, using either MODIFY or the bulk loader.
- **Post-Load** procedures run after the ETL Request loads data into the data target.

For information on the types of tasks you might want to perform with these procedures see *Performing Tasks With Procedures* on page 8-2.

Reference Performing Tasks With Procedures

The following describes the types of tasks you might want to perform with each procedure type:

Pre-Extract

- Connecting to a data source.
- Sending e-mail.
- Performing a specific environment setup.
- Checking for required files.
- Verifying that a table is ready to act as a source.
- Verifying that there is enough space.

- Issuing special connects.
- Allocating files.
- Setting up ETL Manager to transfer files or perform bulk loading.
- Setting a value for an amper variable you might use in an SQL SELECT command.
- Producing a report from a data file or a ETL Manager table, such as the ETL Manager log.

Post-Extract

- Compressing a server transfer file prior to moving it.
- Cleaning up.
- Deleting interim files.
- Copying files to other locations.

Pre-Load

- Preparing a data target for new tables.
- Dropping indexes in the target table.
- Decompressing a file.
- Performing a specific environment setup.
- Checking for required files.
- Verifying that a table is ready to act as a target.
- Verifying that there is enough space.
- Issuing special connects.

Post-Load

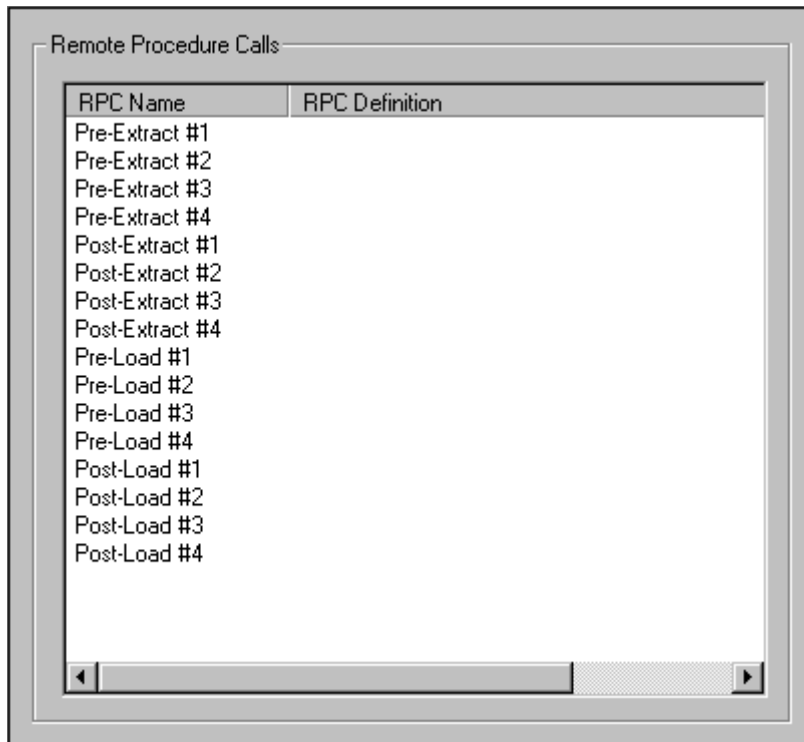
- Cleaning up.
- Deleting interim files.
- Copying files to other locations.

Procedure How to Execute a Procedure With an ETL Request

1. If you are not using a procedure provided by Information Builders, create a procedure in a location accessible by the Server.
2. In the ETL Manager Workbench, open your ETL Request and select the *Procedures* tool. For details on the Procedures tool, see *Procedures Window* on page 8-4.
3. Select the type of procedure you want to run from the RPC Name column.
4. Enter the name of your procedure and any parameters in the RPC Definition column next to the procedure type you selected.

You can pass up to 65 characters to the Server. A procedure name can be up to 8 characters long, and the remaining 57 characters can be used for passing parameters.

Reference Procedures Window



The Procedures window contains the following fields/options:

RPC Name

Are the types of procedures you can supply.

RPC Definition

Is the field in which you enter the name of the procedure you wish to execute.

Executing a Procedure Without an ETL Request

You can run a procedure without executing an ETL Request. This may be useful for testing purposes or for a particular warehouse scenario where the ETL Manager Scheduler is being used, but the only production code is logic within the procedure.

Procedure How to Execute a Procedure Without an ETL Request

1. Create a procedure in an application directory located in APP PATH.
2. In the ETL Manager Workbench, open your ETL Request and select the *Procedures* tool. For details on the Procedures tool, see *Procedures Window* on page 8-4.
3. Select the type of procedure you want to run from the RPC Name column.
4. Enter the name of your procedure and any parameters in the RPC Definition column next to the procedure type you selected.

You can pass up to 65 characters to the Server. A procedure name can be up to 32 characters long, and the remaining 57 characters can be used for passing parameters.

5. In the properties section of the ETL Request main window, check the box labeled *Run Stored Procedures Only (RPCs)*.

Running a Stored Procedure

ETL Manager lets you run any procedure from the Run Stored Procedure window. The procedure is run immediately and the answer set (if any) is returned to the ETL Workbench in the report window.

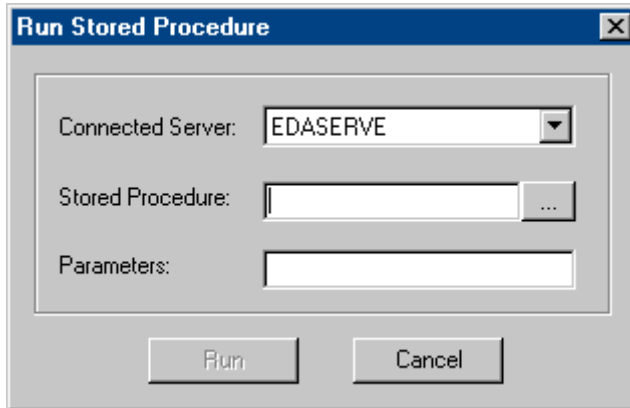
For example, you can create and run a procedure that connects you to a DBMS or produces a report from a data file or an ETL Manager table, such as the ETL Manager log.

Procedure How to Run a Stored Procedure

1. Click *Run Stored Procedures* on the toolbar.



The Run Stored Procedure window opens:



2. Select the server on which the procedure is located from the Connected Server: drop-down list.
3. Enter the name of the procedure in the Remote Procedure: field, or click the ... button to see a list of procedures on the selected server.
4. Enter any necessary parameters in the Parameters: field, separating parameters with a comma.

Note: Running a Remote Procedure in this way runs the procedure immediately and returns the answer set to ETL Workbench. Messages for the procedure will appear in the Message window, depending on the settings specified in the Options window.

Example Displaying All ETL Request Names

1. In the Run Remote Procedure dialog box, enter *CMNAME* in the Remote Procedure: field.
2. Click *Run*.

A list of all ETL Requests appears.

Example Displaying All Table Names

1. In the Run Remote Procedure dialog box, enter *CMGUI SYN* in the Remote Procedure: field.
2. Enter *GET_REMARKS=Y* in the Parameters: field.

This also shows the description for any table that has one.

Viewing Files With EDAGET

You can view a Master File, Access File, or procedure from the Workbench by running the EDAGET procedure.

Procedure How to View a Master File, Access File, or Procedure

1. Select *Run Stored Procedure* from the Tools menu.

The Run Stored Procedure dialog box opens.

2. Enter *EDAGET* in the Stored Procedure field.
3. Enter the following in the Parameters field:

```
{MASTER | ACCESS | FOCEXEC} , name
```

where:

MASTER

Calls a Master file.

ACCESS

Returns an Access File.

FOCEXEC

Returns a procedure.

name

Is the name of the Master File, Access File, or procedure.

The file appears in the Workbench.

Changing Run-Time Behavior With CMOPTION

The CMOPTION procedure, included with the Server, enables you to change the value of a SET parameter or amper variable. If you have one value to test or change, this is the easiest way to perform the operation. It is dynamic and is activated from the ETL Workbench.

Note: You can change the value of a SET parameter by including the parameter in a user profile (EDASPROF), or in a procedure executed from the Procedures tool. You can also change the value of an amper variable using the CMVARS procedure.

Syntax **How to Change the Value of a SET Parameter or Amper Variable With CMOPTION**

In the Procedures tool, in the RPC Definition column, enter

```
CMOPTION {VAR[TABLE]|SET}, name, value
```

where:

SET

Issues a SET command.

VAR

Sets the value of an amper variable.

name

Is the name of the amper variable without the ampersand, or the SET parameter you are changing.

value

Is the value you want to assign to the variable or SET parameter.

Example **Setting the DBA Password With CMOPTION**

The following sets the DBA password to test iWay DBA security settings.

```
CMOPTION SET,USER,password
```

We recommend that you only use the command to *test* the security settings. In a production environment, place the USER setting in an encrypted procedure.

CHAPTER 9

Generating Reports and Viewing the Log File

Topics:

- Running a Report
- Working With a Report
- Viewing the Log File

ETL Manager enables you to view reports on how your ETL Requests run and what error messages the Server generates.

You can also run reports from the iWay Web Console. For more information, see Chapter 12, *Using the iWay Web Console*.

Running a Report

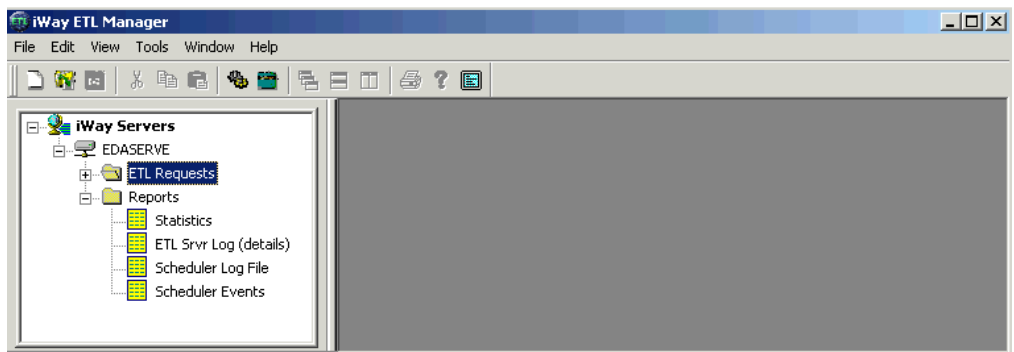
ETL Manager contains several types of reports that enable you to view information about your ETL Requests. The following reports are available:

- The **Statistics** report contains statistical information about the ETL Request, such as how many rows were written, when the ETL Request ran, and how long it took.
- The **ETL Server Log (details)** report contains the message output generated from running the ETL Request. This information includes the steps taken during data extraction and table creation, and any errors that occur.
- The **Scheduler Log File** report displays the output from the Scheduler, the length of time the Scheduler “slept”, and the number of requests that ran.
- The **Scheduler Events** report shows what ETL Requests are scheduled to run on a specified day. The default is to display information about the current date.

Procedure How to Run a Report for All ETL Requests

1. Double-click the *Reports* folder, or right-click the *Reports* folder and select *Explore* from the pop-up menu.

The list of reports expands:



2. Double-click the report you want to run, or right-click the report and select *Open* from the pop-up menu.

The Report Options dialog box opens.

3. Optionally, in the Start Day and Start Time sections, select the earliest date and time on which an ETL Request must have been run to be included in the report.

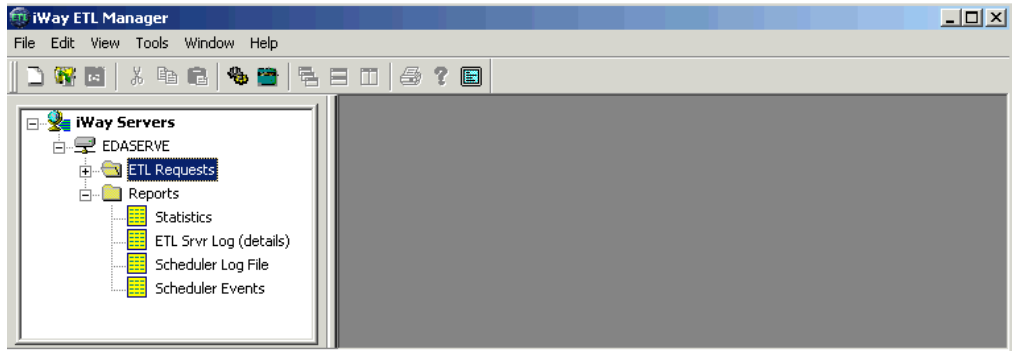
You can click the calendar icon to display a calendar from which to select a date.

4. Click OK.

Procedure How to Run a Report for a Single ETL Request or Job

1. Double-click the *Reports* folder, or right-click the *Reports* folder and select *Explore* from the pop-up menu.

The list of reports expands:



2. Double-click the report you want to run, or right-click the report and select *Open* from the pop-up menu.

The Report Options dialog box opens.

3. Do one of the following:
 - To run a report for a single request, deselect *All ETL Requests*. A list of available ETL Request display. Select an ETL Request from the list.
 - To run a report for a job, enter a job ID in the Job ID field or enter *LAST* to view the most recent execution of the job.

You can find the job ID for an ETL Request in the Statistics report.

4. Optionally, in the Start Day and Start Time sections, select the earliest date and time on which an ETL Request must have been run to be included in the report.

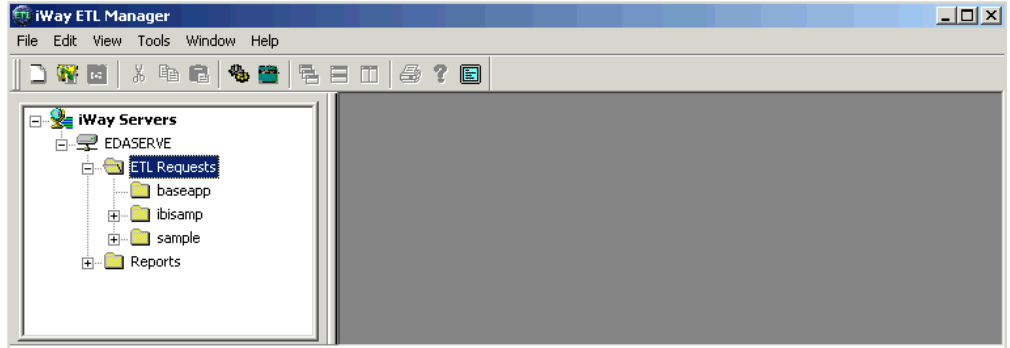
You can click the calendar icon to display a calendar from which to select a date.

5. Click *OK*.

Procedure How to View the Log for a Single ETL Request

1. Double-click the *ETL Requests* folder, or right-click the *ETL Requests* folder and select *Expand* from the pop-up menu.

A list of available applications appears:









2. Double-click the application that contains the ETL Request you are looking for.
3. Right-click the desired ETL Request, and select *View Log* from the pop-up menu to view the log for all executions of the ETL Request, or select *View Last Log* to view the log for the last execution of the ETL Request.

Working With a Report

You can perform tasks such as manipulating the display of a report, copying, saving, or printing a report. These can be performed using the report browser toolbar, the main toolbar, and menus.

Using the Report Browser Toolbar

The buttons on the Report Browser toolbar allow you to work with your report. The toolbar contains the following buttons:

Button	Function
	Highlights your report rows in shades of green and white. This can make reading your report easier, especially if the data stretches across numerous columns.
	Display the report in a spreadsheet grid. Once in spreadsheet format, you can select specific rows to copy into the clipboard.
	Reruns your report. This is useful in case another ETL Request was completed after you ran the original report and you need to update the report data.
	Displays the Report Options dialog box.
	Deletes the report and clears the report browser window. You can also select <i>Clear</i> from the Edit menu.
	Exports your report to a Microsoft Excel spreadsheet.

Note: Some of the functions performed by the Report Browser toolbar buttons can also be performed with the drop-down menus or main toolbar.

Saving a Report

You can save report output to a file.

Procedure How to Save Report Output to a File

1. Select *Save* from the File menu.
The Save As dialogue box opens.
2. Select a location and enter a name for the report.
3. Click *OK*.

Copying a Report

You can copy a report to the clipboard to paste to another program, such as Excel.

Procedure How to Copy Report Output to the Clipboard

1. Click anywhere in the report window.
2. Select *Copy* from the Edit menu.
3. Open the file you want to copy the report contents into, and select *Paste* from that application's Edit menu.

Note: To copy only selected rows or columns to the clipboard, use Spreadsheet format, and select the rows or columns first.

Printing a Report

You can print a report.

Procedure How to Print a Report

Select *Print* from the File menu.

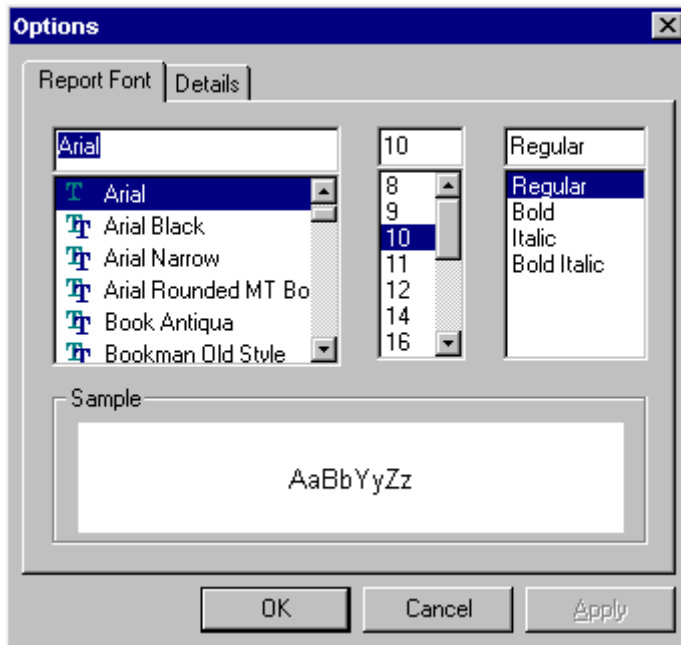
Changing Report Font

You can set the font used in a report. This is useful for report formatting.

Procedure How to Select a Font

1. Select *Options* from the Tools menu.

The Options window opens:



2. Select a font name, size, and attributes and click **OK**.

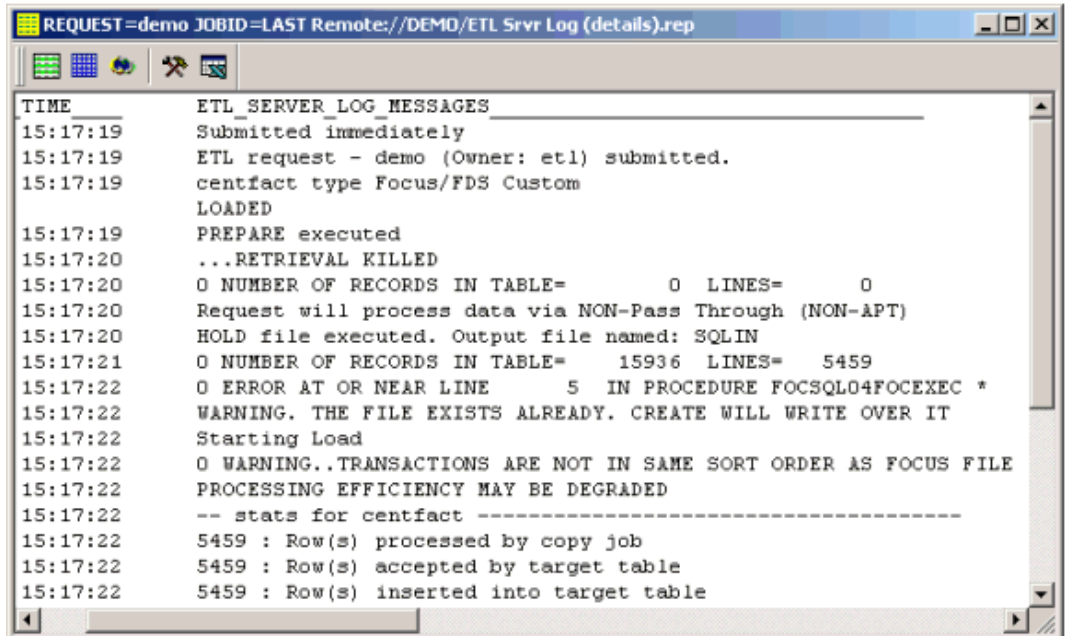
Viewing the Log File

You can view the log file for any ETL Request. This is useful in determining problems encountered when running an ETL Request.

Procedure How to View the Log File

Right-click the ETL Request and select *View Log File* from the pop-up menu, or *View last log* to see only the last log run.

A log file similar to the following appears:



```
REQUEST=demo JOBID=LAST Remote://DEMO/ETL Srvr Log (details).rep

TIME          ETL_SERVER_LOG_MESSAGES
15:17:19      Submitted immediately
15:17:19      ETL request - demo (Owner: etl) submitted.
15:17:19      centfact type Focus/FDS Custom
15:17:19      LOADED
15:17:19      PREPARE executed
15:17:20      ...RETRIEVAL KILLED
15:17:20      0 NUMBER OF RECORDS IN TABLE=      0 LINES=      0
15:17:20      Request will process data via NON-Pass Through (NON-APT)
15:17:20      HOLD file executed. Output file named: SQLIN
15:17:21      0 NUMBER OF RECORDS IN TABLE=    15936 LINES=    5459
15:17:22      0 ERROR AT OR NEAR LINE      5 IN PROCEDURE FOCSQL04FOCEXEC *
15:17:22      WARNING. THE FILE EXISTS ALREADY. CREATE WILL WRITE OVER IT
15:17:22      Starting Load
15:17:22      0 WARNING..TRANSACTIONS ARE NOT IN SAME SORT ORDER AS FOCUS FILE
15:17:22      PROCESSING EFFICIENCY MAY BE DEGRADED
15:17:22      -- stats for centfact -----
15:17:22      5459 : Row(s) processed by copy job
15:17:22      5459 : Row(s) accepted by target table
15:17:22      5459 : Row(s) inserted into target table
```

Procedure How to View the Active Log for an ETL Request

1. In the Browser window, right-click the ETL Request, and click *View Last Log* in the shortcut menu.
2. Click *Run* to refresh the view.
3. The system message will not display until job completion.

Example Sample Server Log File

This example contains sample lines from log files, and an explanation of the code.

The first lines of a log display ETL job initialization.

- This request was submitted with the option of waiting for the ETL Request to complete:
`Submitted immediately`
`ETL request - one (Owner: etl) submitted.`
- This deferred request was submitted without the option of waiting for the ETL Request to complete:
`Submitted for deferred`
`ETL request - two (Owner: etl) submitted`

The next lines display the data target or file type and whether it is a new or existing data target.

- The following is a new table, and ETL Manager attempts to drop and recreate the target table:

```
newtab type MS SQL Server New target
PREPARE executed
Create synonym executed for: test
Issuing DROP TABLE for test
Issuing CREATE TABLE for test
```

- The following is an existing table:

```
oldtab type MS SQL Server Existing target
PREPARE executed
ROWS AFFECTED BY PASSTHRU COMMAND : 222/DELETE
```

The log then displays summary data for the request. You can see the number of records that were processed.

- In this example there were 200 rows processed which ETL Manager attempted to insert into the target table. Of those, 20 were rejected due to duplicate keys and 180 were accepted by the data target.

```
Starting Load-- stats for newtab -----
200 : Row(s) processed by ETL job
180 : Row(s) accepted by target table
200 : Row(s) inserted into target table
0 : Row(s) updated in target table
0 : Row(s) deleted from target table
0 : Row(s) rejected due to format errors
0 : Row(s) rejected due to validation errors
0 : Row(s) rejected due to no match logic
20 : Row(s) rejected because duplicate exists
```

The remaining lines contain summary data, and the return code from the ETL Request.

- In the following, a return code of zero (0) means the request completed successfully:

```
Ending Load
ETL Return Code = 0
ETL request: one - finished processing
Finished
```

CHAPTER 10

Data Management

Topics:

- Storing ETL Requests
- Application Directories
- Using a Data Source
- Using a Data Target
- Using a Synonym

This topic describes the data sources, data targets, and Synonyms used in ETL Manager.

Storing ETL Requests

ETL Requests are stored on the Server. ETL manager stores all requests on the Server.

Each ETL Request is stored as two files:

- A ready to run procedure with the file extension .fex.
- A file containing detailed information needed to edit the ETL Request from the Workbench. This file has an extension of .etl.

Application Directories

ETL Manager uses Server application name spaces. This affects:

- The directories where ETL requests are stored.
- The synonyms available to the ETL Manger user.

By default, ETL Requests and synonyms are stored in the BASEAPP directory. To save requests in another directory, the directory must be:

- Created or mapped under APPROOT.
- Added to the application path in the server or user profile.

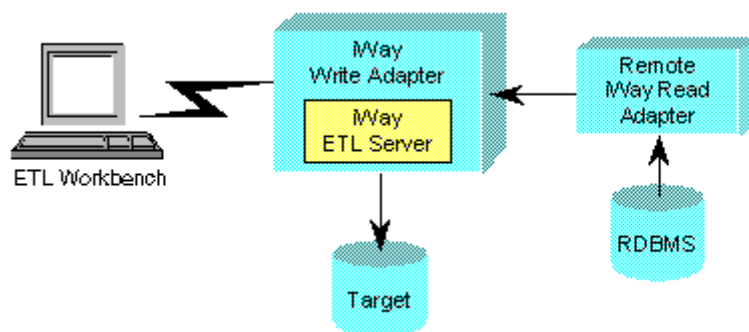
These steps can be done in the Web Console. For more information on these procedures as well as application directories, see *Configuring the Application Path* in Chapter 12, *Using the iWay Web Console*.

Using a Data Source

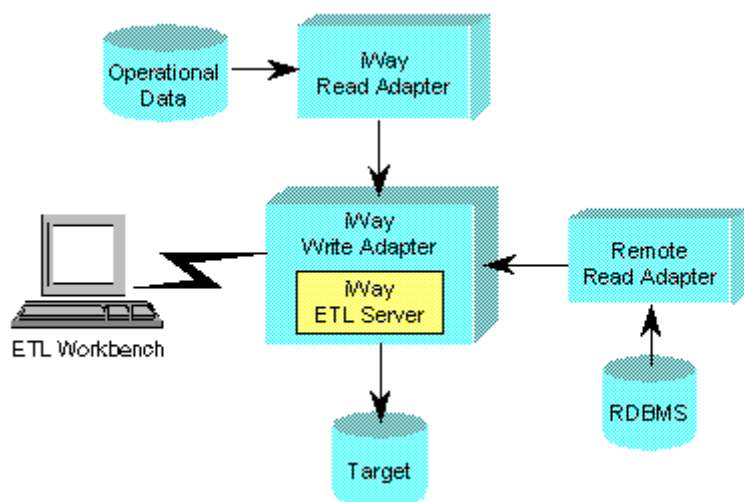
You can use ETL Manager to access data sources across an enterprise and copy those data sources into various data targets. The Server accesses any iWay-supported data source, which includes numerous formats on over 35 platforms.

The Server establishes a connection with a remote server to access this data. In the simplest case, the Server may access a single source. Data can be accessed in two ways:

- From a remote Server. The following figure illustrates how data can be migrated to the Server from a remote Server:



- From the Server. Because the Server utilizes iWay Hub Services, cross-platform joins can be performed when accessing data in this way. The following figure illustrates this process:



Naming a Data Source

A data source name must conform to the naming conventions of your operating system. When naming a data source, you must avoid the following:

- A name beginning with the letters CM, ETL or SYS. These prefixes are reserved for ETL Manager internal stores and the tables in the iWay Catalog.
- In a FOCUS or Fusion data source, a non-unique field name.
- Names that are used as commands within ETL Manager. These are HOLD, SQLIN, and SQLGET.
- Any name with special characters such as *, &, \$, %, @, _ or blanks.

Using a Data Target

ETL Manager copies the data from a data source into a data target.

- Local relational data target. For details see *Local Relational Data Targets* on page 10-5.
- Local non-relational file. For details see *Local Non-Relational Files* on page 10-6.
- Server transfer file. For details see *Server Transfer Files* on page 10-6.
- Remote destination. For details see *Remote Destination* on page 10-7.

Reference Special Notes for Oracle Users

ETL Requests that load data into ORACLE tables with columns described as VARCHAR2 (variable length character strings) may fail with the following message:

`ORA-01400: mandatory (NOT NULL) column is missing or NULL during insert`

This message occurs when a variable length character column in the target is described as NOT NULL and the source column contains blanks, as is often the case if the source is a legacy data source or a flat file.

By default, when ORACLE loads data into a VARCHAR2 column, trailing spaces are dropped. If the source column only contains spaces, the result is a NULL.

There are several different ways to address this issue. They are listed in order of preference:

- Do not describe the target columns as NOT NULL. If a column must be unique, when creating the table, describe the column with CONSTRAINT name UNIQUE instead of describing the column as a PRIMARY KEY.
- Use a DEFINE in the Master File for the source data that ensures no blanks are ever retrieved for the column by using some default value. For example:

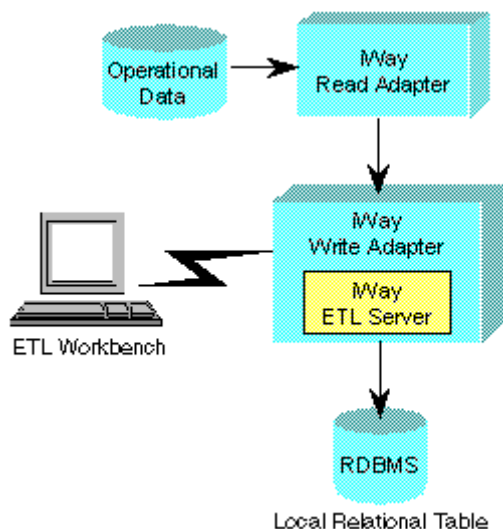
```
COL/A5 = IF COL EQ ' ' THEN '~' ELSE COL;
```

- Pad the character column with spaces by:
 1. Opening the *Target* tool.
 2. Selecting the *Output Parameters* tab.
 3. Selecting the value *FIX* for ORACLE Blank Column.

This allows spaces to be written to the VARCHAR2 columns by padding the column with trailing blanks to the full width of the column. However since all values are padded with blanks, a WHERE test against the column won't work as expected.

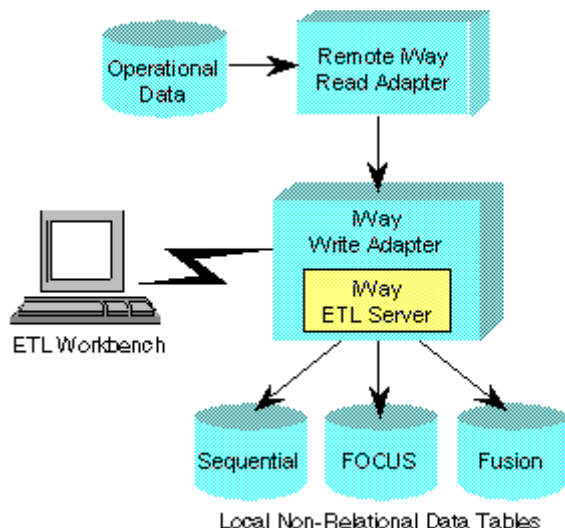
Local Relational Data Targets

Depending on the platform, supported relational data targets currently include the following: DB2, Oracle, Informix, Microsoft SQL Server, Sybase, MS Access, Nucleus, NCR Teradata, and ODBC. The following illustrates ETL Manager creating a data warehouse in a local RDBMS:



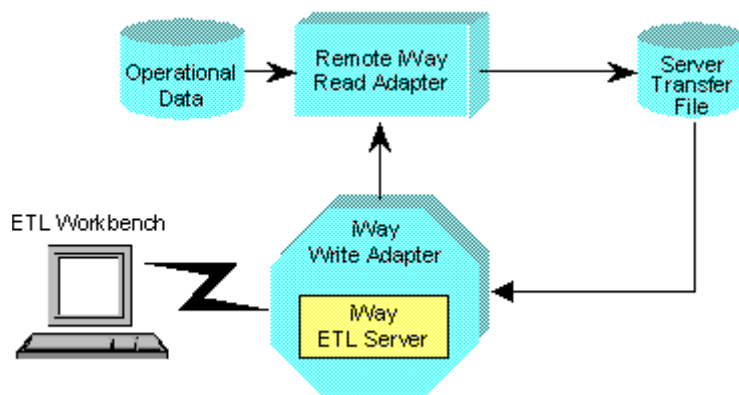
Local Non-Relational Files

A flat sequential file, FOCUS, or Fusion data target can be created on the Server. A flat sequential file is created in the character set appropriate for that platform (for example, ASCII on UNIX or EBCDIC on OS/390). This process is illustrated in the following:



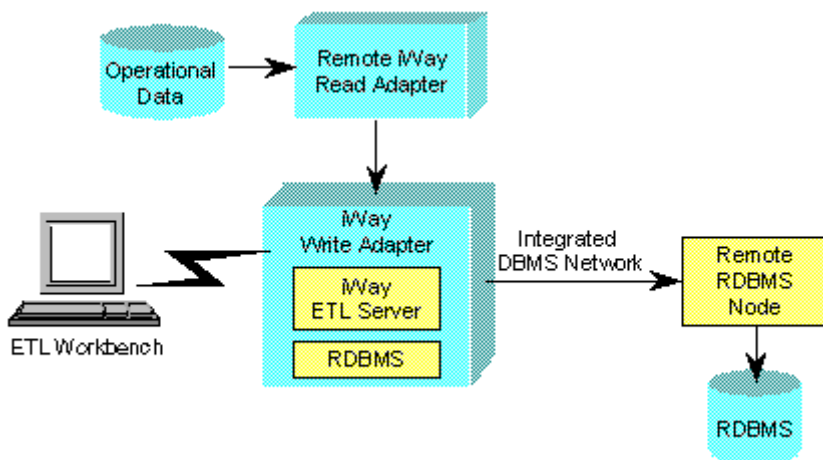
Server Transfer Files

You can create a Server Transfer File (flat sequential file) on the source Server. This is useful when some other mechanism such as FTP (or IND\$FILE for OS/390) is used to send the file to another platform. You would use this option if you wanted to migrate large quantities of data as quickly as possible. The following illustrates this process:



Remote Destination

Many relational data source management systems provide for remote table access. If properly configured, Server supports the syntax necessary to write to remote targets (for example, SQL*NET provides access to remote Oracle tables). The following figure illustrates this process:



Naming a Data Target

ETL Manager supports up to 32 characters for the data target name. If you are loading to a relational data target, longer names are possible because of the addition of data source and owner details. However, the final table name known to the iWay Catalog remains limited to 32 characters.

When naming a data target, you must avoid the following:

- A name beginning with the letters CM, ETL, or SYS. These prefix values are reserved for ETL Manager internal tables and the tables in the iWay Catalog.
- In a FOCUS or Fusion data source, a non-unique field name.
- Names that are used as commands within ETL Manager. These are HOLD, SQLIN, and SQLGET.
- Any name with special characters such as *, &, \$, %, @, _, or blanks.

Naming Target Columns

Most column names are allowed. However, ETL Manager uses a notation for columns of an answer that could conflict with names in your warehouse target. Avoid the following column names:

- Names beginning with the capital letter E followed by a two or three digit number, for example, E01 - E99.
- Names beginning with the capital letter T followed by a number, for example, T1 - T16.
- Names containing spaces, dashes, and other non-standard characters in both the source and target data columns (*, &, #, \$, %, @, _).

Using a Synonym

A Synonym (Master File) contains metadata about a data source. It consists of attributes that describe a data source. A Master File and the data source that it describes usually have the same name. The following conditions should be avoided in your iWay Synonyms:

- Column names with special characters. See *Naming Target Columns* on page 10-8.
- Duplicate column names. Care can be used if the duplicate names are not referenced (that is, FILLER).

There are various limits within the Server that may affect ETL Manager's ability to retrieve and load data to target data marts. These limits differ from server to server and are affected by available memory, number of columns, length of column names, and overall LRECL of a legacy table.

The source and target objects display all the synonyms on the server in the BASEAPP directory plus the directories in the current application path. To see the current application path in the request browser, right click on the server and select *Properties*.

Reference Guidelines for Synonyms

Use the following general guidelines when creating your Synonyms.

- File definitions must not exceed any of the following:
 - Approximately 3000 columns.
 - 30K of total column name character length.
 - 16K for single LRECL (total length of field formats, one instance of a row).

See your *iWay Data Administration Guide* for suggestions on how to access your structures without pressing these limits. Some strategies for doing this are using OCCURS clauses, separating Synonyms for different record types, using filler fields, and so on.

- Multiple synonyms with the same name should be avoided.

If multiple synonyms with the same name exist in different application directories, the synonym located in the directory that is higher in the search order is used. For example, if you are using the TEST synonym and the user's profile contains the line

APP PATH RED BLUE

ETL Manager finds synonyms named TEST in the RED, BLUE and BASEAPP directories. The TEST synonym located in the RED directory is used because it is the directory that appears first in application path.

Using a Synonym

CHAPTER 11

ETL Manager Security

Topics:

- ETL Request Execution and User IDs
- Data Targets and User Access
- Restricting the Application Paths Available to a User

This topic describes issues related to establishing security for ETL Manager.

ETL Request Execution and User IDs

The iWay agent created when you connect to a Server has an associated logon user ID. Local file, directory, and resource security is controlled by that user ID. If additional security is needed when the Server connects to a remote source or relational subsystem, it is specified in the EDASPROF file. Trusted node, password pass-through, or explicit verification is used; see the iWay Server manual for your platform for additional details. At run time, an ETL Request is assigned a user ID and behaves like a normal iWay user.

Note: All ETL requests run with the server user ID.

Request Submission

Any user can submit an ETL Request located in the server application path.

Users without individual profiles can submit requests in the application directories in the server profile.

Users with individual profiles can submit requests in the application directories in their individual user profiles.

Data Targets and User Access

ETL Manager is typically used to create data targets in the local relational subsystem. This requires that the user ID processing ETL Requests has appropriate access to the target RDBMS. If the request's data target is a new table, the user ID must have permission to issue CREATE TABLE statements. If the request's data target is an existing table, transaction update access is necessary. Relational data targets created in ETL Manager are owned by the user ID specified in the connection string in the server profile.

Where flat files or Server Transfer Files are being created, the user ID of the processing request must have access to the subdirectory or data set that is currently allocated to receive the answer set. Flat files created in ETL Manager are owned by the server user ID.

Restricting the Application Paths Available to a User

By default, the system profile (EDASPROF.PRF) is run for all users when they connect to the Server, so users have access to all application paths. You can control a user's access to application directories by creating individual user profiles. Each user can then access only the application directories specified in the application path in their individual profile. Each user can only use synonyms in the specified application path.

Note: User profiles are only used if server security is enabled.

Procedure **How to Set the Application Paths Available to a User**

1. From the Web Console, click the *ETL* link.
The ETL Configuration window opens.
2. Click the *Configure Application Path* link.
The Configure Application Path window opens.
3. Do one of the following:
4. Select a user from the Selected User drop-down list, or select *--New User--* to set restrictions for a new user and enter the new user name in the User name field.
5. Click *Edit*.
The Edit User Profile window opens.
6. Ensure that the profile contains the following line:
`APP ENABLE`
7. Edit the application path to include the desired application directories viewable for the selected user ID:
`APP PATH dir1 dir2 dir3 ...`
8. Click *Save*.

CHAPTER 12

Using the iWay Web Console

Topics:

- Starting ETL Manager
- Configuring ETL Manager Behavior
- Configuring the Application Path
- Controlling the ETL Scheduler
- Refreshing Information in the Web Console
- Working With ETL Requests in the Web Console

You can configure ETL Manager behavior in the Web Console. The Web Console allows you to do such things as view and run procedures, configure the application path, and set ETL Manager variable values.

Starting ETL Manager

You can configure ETL Manager using the ETL section of the iWay Web Console. The Web Console can only be accessed after the Server has been started.

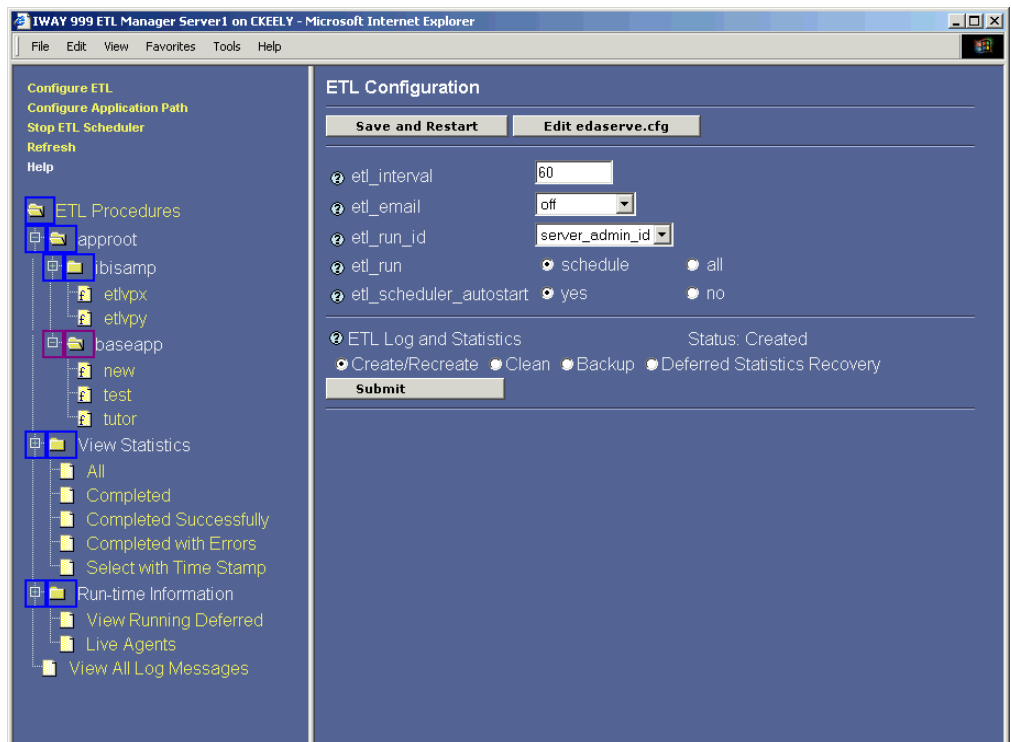
Procedure How to Open the ETL Configuration

1. Ensure the Server is running.
2. In the Start Menu, select *Programs*, then the *iWay 52 ETL Manager Server* program folder, then *Web Console*.

The Web Console opens in your browser.

3. Click the *ETL* link.

The ETL window opens with the ETL Configuration window displaying:



4. Select an option from the left frame. The options are:

Configure ETL opens the ETL Configuration window. For details, see *Configuring ETL Manager Behavior* on page 12-3.

Configure Application Path opens the Configure Application path window. For details, see *Configuring the Application Path* on page 12-7.

Stop ETL Scheduler stops the ETL Scheduler from checking for active ETL Requests. For details, see *Controlling the ETL Scheduler* on page 12-10.

Stop All Running Procedures cancels the execution of all procedures currently running. For details, see *Cancelling ETL Requests* on page 12-10.

Refresh refreshes the information displayed in the left frame of the Web Console.

Help opens online help.

ETL Procedures tree displays all the ETL Requests and directories available on the Server. For details, see *Working With ETL Requests in the Web Console* on page 12-10.

Configuring ETL Manager Behavior

You can control the behavior of ETL Manager, the Scheduler, and internal tables with the Web Console. You can do the following:

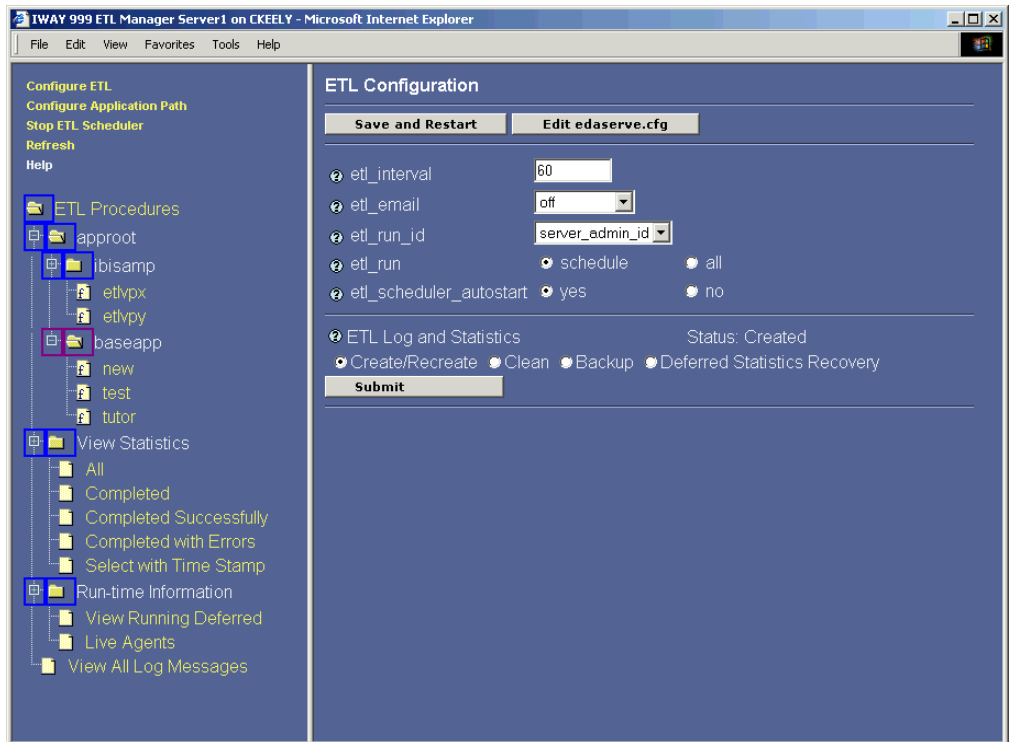
- Set Scheduler settings.
- Set e-mail preferences.
- Set up security.
- Work with log and statistics tables.
- Edit the edaserv.cfg file.

For details on these settings, see *ETL Configuration Window* on page 12-5.

Procedure How to Configure ETL Manager Behavior

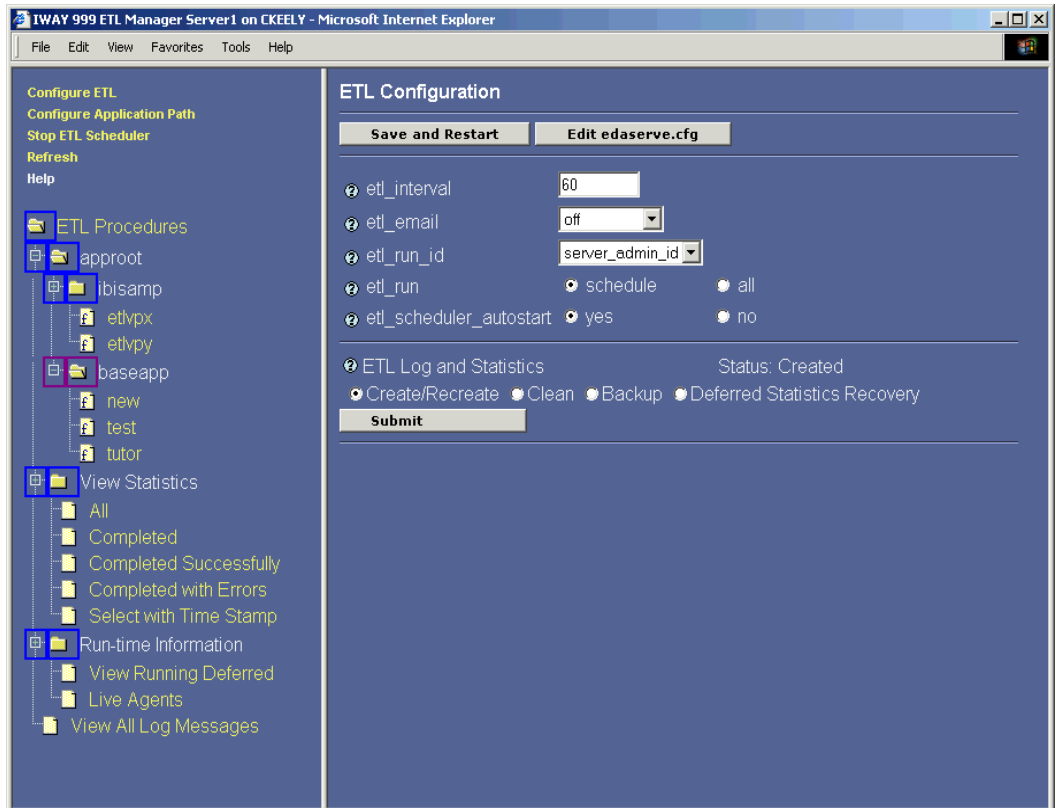
1. Click the *Configure ETL* link.

The ETL Configuration window opens:



2. Make your changes, and click *Save and Restart*. For details on the options found in the ETL Configuration window, see *ETL Configuration Window* on page 12-5.

Reference ETL Configuration Window



The ETL Configuration window contains the following fields/options:

Save and Restart

Saves your changes and restarts the Server.

Edit edaserv.cfg

Allows you to view and edit the edaserv.cfg file in the Web Console.

etl_interval

Is how often, in seconds, you want the Scheduler to check for ETL Requests. The default value is 60 seconds.

etl_email

Indicates when to send e-mail for all ETL Requests. The options are:

off does not send e-mail. This value is the default.

completion sends e-mail when every ETL Request has completed.

failure sends an e-mail when an ETL Request fails.

You can also set ETL Manager to use e-mail notification by selecting the *Notify* check box in the Properties panel of your ETL Request.

Before using e-mail notification, you must configure the server for it. For details, see the Installation and Configuration manual for your platform.

etl_run

Determines whether to run ETL Requests that are specified as Scheduled, or to run all ETL Requests. The options are:

schedule runs ETL Requests at their scheduled time. This is the default value.

all runs all ETL Requests immediately regardless of their scheduled execution time.

etl_share

Allows users to access all ETL Requests regardless of who saved them to the server. The options are:

on allows users to access all ETL Requests. This is the default value.

off allows users to access only ETL Requests which they saved to the server.

etl_scheduler_autostart

Specifies whether the Scheduler should check for ETL Requests automatically, or only when prompted. The options are:

yes automatically starts the Scheduler when the Server starts. This is the default value.

no does not start the Scheduler when the Server starts.

ETL Log and Statistics

Specifies the action to take on the log and statistics tables. The options are:

Create/Recreate creates or recreates empty log and statistics tables. This is the default value.

Clean deletes all rows from the log and statistics tables up to a specified date.

Backup creates a backup of the log and statistics tables named ETLLOGBK and ETLSTABK, respectively.

Deferred Statistics Recovery moves information about deferred ETL Requests to the log and statistics tables if the internal tables become corrupted.

Submit

Submits your choice for the action to take on the log and statistics tables.

Configuring the Application Path

You can determine the application logic for ETL Manager in the Web Console. ETL requires APP PATH to be enabled.

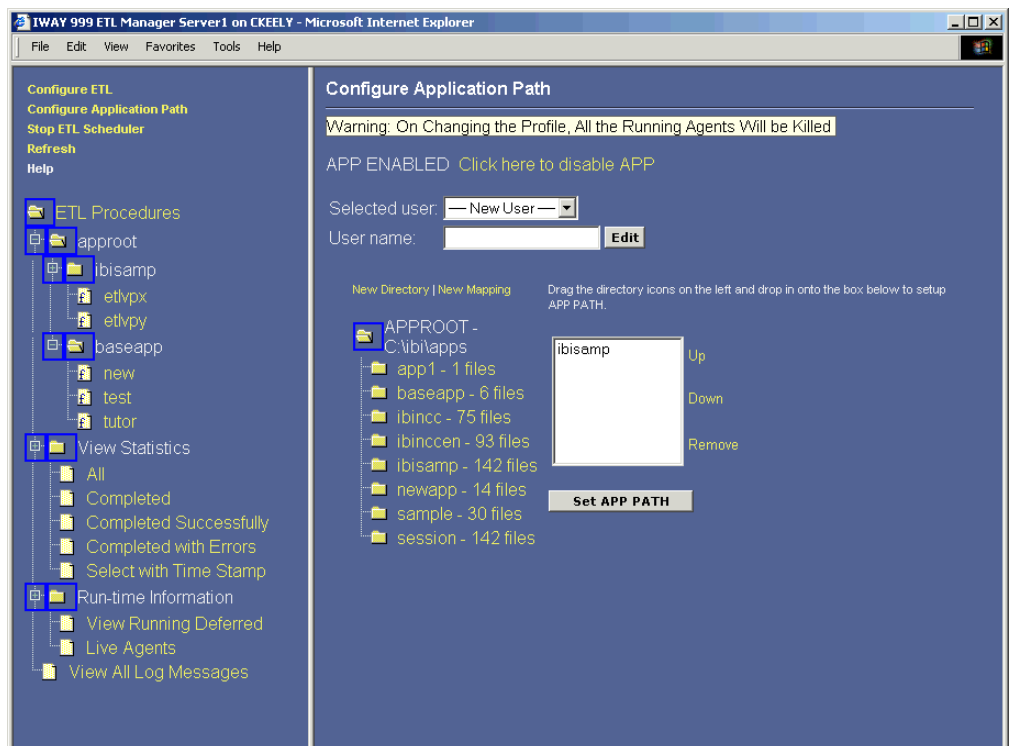
APP PATH is a list of subdirectories beneath a common parent directory called APPROOT. APP PATH is the default search criteria for the Server. The Server searches each of the directories listed in APP PATH sequentially to find the target procedure. If APP PATH logic is enabled, it disables EDAPATH. APP PATH logic can be issued from a procedure, a profile, or from an HTML launch page.

You can also use the Configure Application Path window to set the viewable directories for ETL Manager, as well as add new directories and mappings to APPROOT. To use ETL Requests that reside in a directory other than ibi\apps\baseapp, the directory must be added to the list of viewable directories.

Procedure How to Configure the Application Path

1. In the ETL Configuration window, click the *Configure Application Path* link.

The Configure Application Path window opens:



2. Make your changes, and click *Set APP PATH*. For details on the options in this window, see *Configure Application Path Window* on page 12-8.

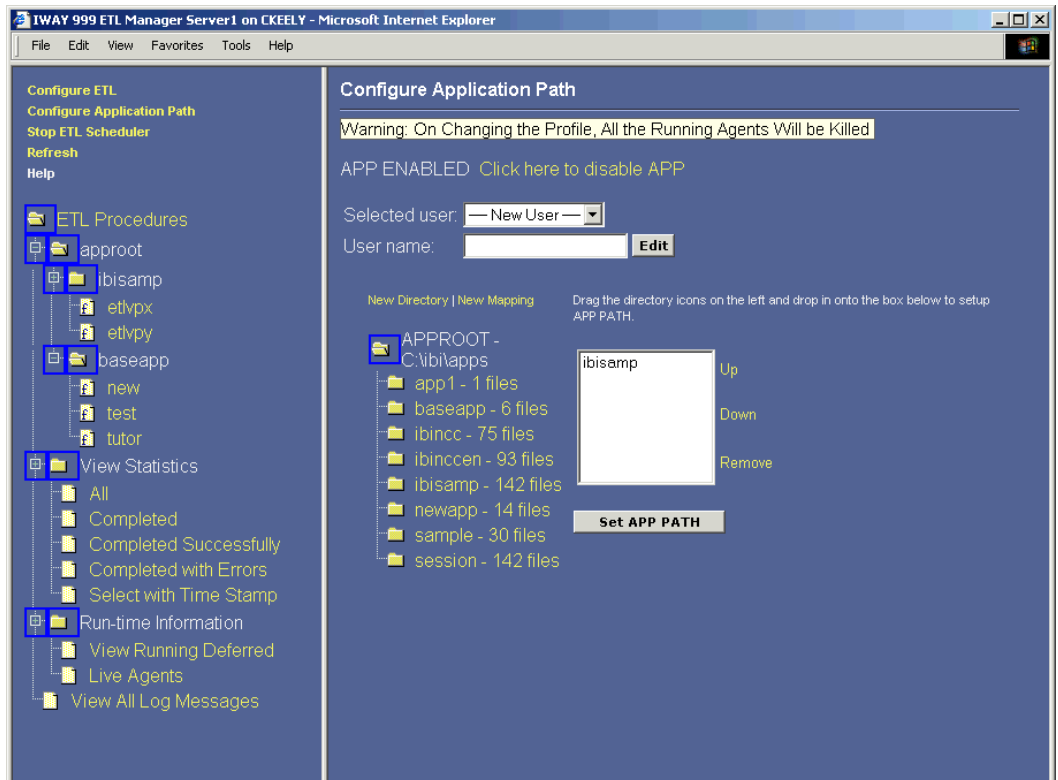
Procedure How to Create a New Application Directory

1. In the Configure Application Path window, click *New Directory*.
2. Enter the name of the new directory, and click *OK*.

Procedure How to Set the Server Application Path

1. In the APPROOT directory tree, drag and drop the directory you want to add from the list of directories to the APP PATH window.
2. Set the directory search order by moving the directories into their search order using the *Up*, *Down*, and *Remove* links.
3. Click *Set APP PATH*.

Reference Configure Application Path Window



The Configure Application Path window contains the following fields/options:

APP ENABLED / APP DISABLED

Specifies whether APP PATH is enabled or disabled. ETL Manager requires that APP PATH be enabled.

Click here to disable APP / Click here to enable APP

Allows you to enable and disable APP PATH.

Selected User

Is the user for whose profile is affected by the APP PATH information.

User name

For a new user, is the name of the user who is affected by that APP PATH information.

New Directory

Creates a new directory in APPROOT.

New Mapping

Allows you to add a new mapping to APPROOT.

APPROOT directory tree

Displays the directories within APPROOT.

APP PATH window

Allows you to specify the directories viewable from ETL Manager. You specify the directories to be included by dragging them from the APPROOT directory tree into the window.

Up

Moves the selected directory up in the list of directories. The order of the directories determines what directories are searched first.

Down

Moves the selected directory down in the list of directories. The order of the directories determines what directories are searched first.

Remove

Removes a directory from the list of directories viewable from ETL Manager.

Set APP PATH

Saves the changes made to the APP PATH settings.

Controlling the ETL Scheduler

You can turn the ETL Scheduler on and off using the ETL section of the Web Console. When the scheduler is on, a link displays that allows you to stop the Scheduler. When the Scheduler is turned off, a link displays that allows you to start the Scheduler.

Procedure How Stop the ETL Scheduler

Click the *Stop ETL Scheduler* link.

Procedure How Start the ETL Scheduler

Click the *Start ETL Scheduler* link.

Cancelling ETL Requests

You can stop the execution of ETL Requests from the Web Console.

Procedure How to Cancel the Execution of ETL Requests

Click the *Stop All Running Procedures* link.

Refreshing Information in the Web Console

You can refresh the Web Console in order to reflect any changes made to the Server.

Procedure How to Refresh the Information in the Web Console

Click the *Refresh* link.

Working With ETL Requests in the Web Console

You can work with ETL Requests in the Web Console using the procedures directory tree.

The ETL procedures directory tree contains a list of all the directories available on APP PATH and the ETL Requests residing within them. You can use the procedures directory tree to do the following:

- View the application directory structure and the location of ETL Requests.

- Work with ETL Requests in the following ways:
 - View the ETL Request. For details, see *Viewing an ETL Request* on page 12-11.
 - Run the ETL Request. For details, see *Running an ETL Request* on page 12-11.
 - Run the ETL Request as deferred. For details, see *Running an ETL Request as Deferred* on page 12-11.
 - Delete the ETL Request. For details, see *Deleting ETL Requests* on page 12-12.
 - View the log or last log for the ETL Request. For details, see *Viewing an ETL Request's Log* on page 12-13.
- View statistics and message logs. For details, see *Viewing the Statistics and Message Logs* on page 12-14.

Viewing an ETL Request

You can view an ETL Request's code in the Web Console.

Procedure How to View an ETL Request

1. Click the name of the ETL Request you want to view. A pop-up menu opens.
2. Click *View Procedure*.

The procedure's code opens in the Web Console.

Running an ETL Request

You can run an ETL Request from the Web Console. When an ETL Request is run from the Web Console, the results and any messages display in the Web Console.

Procedure How to Run the ETL Request

1. Click the name of the ETL Request you want to run. A pop-up menu opens.
2. Click *Run Procedure*.

The ETL Request executes, and the results appear in the Web Console.

Running an ETL Request as Deferred

You can specify an ETL Request to run as deferred from the Web Console. When this is done, a confirmation message and any error messages display in the Web Console.

Procedure How to Specify an ETL Request as Deferred

1. Click the name of the ETL Request you want to run. A pop-up menu opens.
2. Click *Run Deferred*.

The ETL Request is added to the queue of deferred Requests, and a confirmation message appears in the Web Console.

Deleting ETL Requests

You can delete one or several ETL Requests from the Web Console. If you delete an ETL Request from the Web Console, it will no longer appear in the application folder in the Workbench.

Procedure How to Delete an ETL Request

To delete a single ETL Request:

1. Click the name of the ETL Request you want to delete. A pop-up menu opens.
2. Click *Delete Procedure*.

To delete multiple ETL Requests:

1. Click on the page icon next to each procedure you want to delete.
A check marks each of these procedures.
2. Click the name of one of the marked procedures. A pop-up menu opens.
3. Click *Delete Procedure*.

Viewing ETL Request Run-Time Information

You can view run-time information for an ETL Request from the Procedures tree. You can view:

- Deferred request information. For details, see *How to View Deferred Request Information* on page 12-12.
- Agent information. For details, see *How to View Agent Information* on page 12-13.

Procedure How to View Deferred Request Information

1. In the Procedures tree, expand the *Run-time* information folder.
2. Click *View Running Deferred*.

The results appear in the Web Console.

Procedure How to View Agent Information

1. In the Procedures tree, expand the *Run-time* information folder.
2. Click *Live Agents*.
The Agents window opens.
3. Select the information you want to view:
Statistics displays statistics for the live agents.
Agents displays details for the active agents.
Sessions displays the agents currently in a session.
Connections displays the connection the agents are using.
Deferred statistics displays information for all deferred requests for the agents.
Deferred list displays a list of deferred requests.
4. Optionally, select the *Refresh* checkbox and specify the number of seconds in which you want the information refreshed.

Viewing an ETL Request's Log

You can view an ETL Requests log of last log from the Web Console.

- If you view an ETL Request's log, all information for an ETL Request is retrieved.
- If you view an ETL Request's last log, all information for the last execution of an ETL Request is retrieved.

Procedure How to View an ETL Request's Log

1. Click the name of the ETL Request whose log you want to view. A pop-up menu opens.
2. Click *View Log* or *View Last Log*.

The selected log appears in the Web Console.

Viewing the Statistics and Message Logs

You can view the statistics and message logs from the Web Console. The statistics and message logs display all messages sent to ETL Manager for a specific ETL Request.

Procedure How to View Statistics and Message Logs With the Web Console

In the View Statistics folder, click a selection. The options are:

All displays statistical information about all ETL Requests.

Completed displays statistical information about all completed ETL Requests.

Completed Successfully displays statistical information about all ETL Requests that have completed successfully.

Completed With Errors displays statistical information about all failed ETL Requests.

Incomplete or View Running displays statistical information about all deferred ETL Requests.

Select with Time Stamp displays statistical information about all ETL Requests run during a specified period of time.

View All Log Messages displays the run-time message log.

Migrating ETL Requests

You can migrate ETL Requests when you upgrade ETL Manager to a newer version. To migrate ETL Requests, you must do the following:

1. Migrate Master Files and Access Files. For details, see *How to Migrate Master Files and Access Files* on page 12-15.
2. Prepare for the migration of the ETL Requests. For details, see *How to Prepare for Migration On OS/400* on page 12-16, *How to Prepare to Migrate ETL Requests Stored in a Relational Data Source on Windows NT/2000/XP or UNIX* on page 12-16, and *How to Prepare to Migrate ETL Requests Stored in FOCUS* on page 12-17.
3. Migrate the ETL Requests. For details, see *How to Migrate ETL Requests* on page 12-17.

Procedure How to Migrate Master Files and Access Files

1. Locate the Master Files and Access files used by your ETL Requests. The following are the default locations for each operating system:
 - **Windows NT/2000/XP**
Version 4.3: c:\ibi\srv43\cpm\catalog
Version 5.1: c:\ibi\srv51\etl\catalog
 - **UNIX**
Version 4.3: home/ibi/srv43/cpm/catalog
Version 5.1: home/ibi/srv51/etl/catalog
 - **OS/400:** /QSYS.LIB/EDAMAIN.LIB
2. Use the operating system copy command to copy the files to the location of the Release 5.2 ETL Requests directory. The default locations in Release 5.2 are:
 - **Windows:** c:\ibi\apps\baseapp
 - **UNIX:** home/ibi/apps/baseapp

Master Files and Access Files can also be copied to other locations:

- To use other directories under ibi\apps, these directories must be added to the server's application path. For details, see *Configuring the Application Path* on page 12-7.
- To use directories that are not under the ibi\apps directory, the directories must be mapped and added to the application path. See the server configuration manual for your platform for details.

Procedure How to Prepare for Migration On OS/400

1. Copy the metadata for the CMSCHED table from the QSYS library (the default location for this library is /QSYS.LIB/EDAMAIN.LIB/MASTER.FILE) to the IFS directory. To do this, in an OS/400 session, enter the following command to copy the CMSCHED Master Files and Access files:

To copy the Master file

```
CPYTOSTMF FROMMBR('/QSYS.LIB/EDAMAIN.LIB/MASTER.FILE/CMSCHED.MBR')  
TOSTMF('/home/ibi/srv52/etl/etlmgr/cmsched.mas')
```

To copy Access file:

```
CPYTOSTMF FROMMBR('/QSYS.LIB/EDAMAIN.LIB/ACCESS.FILE/CMSCHED.MBR')  
TOSTMF('/home/ibi/srv52/etl/etlmgr/cmsched.acx')
```

With the “replace” option

```
CPYTOSTMF FROMMBR('/QSYS.LIB/EDAMAIN.LIB/ACCESS.FILE/CMSCHED.MBR')  
TOSTMF('/home/ibi/srv52/etl/etlmgr/cmsched.acx') STFMOPT(*REPLACE)
```

2. After the Master Files and Access Files have been copied, edit CMCSHED.ACX and change the file name so that it points to the physical location of the table. This is done with the following:

```
from FILENAME=CMSCHED  
to FILENAME=EDAMAIN/CMSCHED
```

where:

EDAMAIN

Is the name of the library where the DB2 table resides.

Procedure How to Prepare to Migrate ETL Requests Stored in a Relational Data Source on Windows NT/2000/XP or UNIX

Configure a data adapter for the data source, such as ORACLE, MS SQL Server or DB2. For more information, refer to the server configuration manual for your platform.

Procedure How to Prepare to Migrate ETL Requests Stored in FOCUS

1. The migration procedure requires that the CMSCHED.FOC and CMSCHED.MAS files be located in the same directory. In the Windows NT/2000/XP ibi directory or the UNIX home directory, copy the CMSCHED.FOC file to the directory containing CMSCHED.MAS For example:

- **Windows NT/2000/XP**

Version 4.3: from c:\ibi\srv43\cpm\fds to c:\ibi\srv43\cpm\copymgr

Version 5.1: from c:\ibi\srv51\etl\fds to c:\ibi\srv51\etl\etlmgr

- **UNIX**

Version 4.3: from home/ibi/srv43/cpm/fds to home/ibi/srv43/cpm/copymgr

Version 5.1: from home/ibi/srv51/etl/fds to home/ibi/srv51/etl/etlmgr

Procedure How to Migrate ETL Requests

1. Click on the *ETL* link in the list on the left of the window.
The ETL Configuration window opens.
2. Click on the *ETL Procedures* folder on the left of the window.
A context menu opens.
3. Click on *Migrate 4.x Procedures*.
The Migrate window opens.
4. Enter the path to the directory where CMSCHED.MAS and CMSCHED.ACX reside on the 4.x or 5.1.x Server. The default locations for each platform are as follows:
 - **Windows NT/2000/XP**
Version 4.3: c:\ibi\srv43\cpm\copymgr
Version 5.1: c:\ibi\srv51\etl\etlmgr
 - **UNIX**
Version 4.3: home/ibi/srv43/cpm/copymgr
Version 5.1: home/ibi/srv51/etl/etlmgr
 - **OS/400:** Use the location specified in the previous procedure.
5. Click *Submit*.
The message "Migrate procedure submitted" appears.

6. When the migration procedure finishes, review the log created to ensure that all ETL Requests migrated successfully. The following message should appear for each ETL Request:

(ICM18164) Request NAME was created/updated successfully for user: USER.

The following message should also appear:

(ICM18354) Migration procedure completed successfully.

For other messages refer to *Server Message Codes* on page B-6.

7. Review the list of new application directories. By default, ETL Requests are located in the \ibi\apps\baseapp directory. If a category name is assigned in the Properties window, a directory with that category's name is created under \ibi\apps and the following message appears:

```
*----- New Created APP directories -----*/  
newapp  
*-----*/
```

Note: To view or run ETL Requests in these new application directories they must be added to the server application path. See *Configuring the Application Path* on page 12-7 for details.

CHAPTER 13

Improving Performance

Topics:

- Designing Your Application
- Reviewing Your Data Sources
- Reviewing Data Targets
- Customizing Your Configuration
- Prototyping
- Improving ETL Manager's Performance
- Customizing Your Environment
- Creating a DEFINE in the Master File

Building a data warehouse involves the integration of many technologies and disciplines. The configuration and data type combinations are endless, as are the volumes of data and frequency of migration. Like any application, there is no substitute for careful planning and thorough design.

Considering the issues and questions in this topic will lead to the successful deployment of your data mart and data warehouse implementations using ETL Manager and the iWay architecture.

Designing Your Application

Data warehouses and/or data marts are created for a variety of reasons. Knowing the factors leading to the decision for building an extract will help you identify the issues for ETL management that will have the greatest impact on your success. The ultimate goal is always to have an automated process that moves and converts all of your data in a very small period of time. However, it helps to know what issues are most important in your company's criteria for a successful data warehouse. Are you consuming large amounts of time in the management of 3GL extraction programs and load routines? Are you uncomfortable with ad hoc user access to production legacy systems? Is your network unable to handle the performance requirements for decision support and analysis?

If you consider the reasons for warehousing, your attention will be properly focused on the critical variables. ETL Manager provides you with the flexibility to control the variables inherent at your site and in your data.

From an application perspective, you need to evaluate what will direct your overall data movement strategy. Considering the following questions will aide in this as you learn more about ETL Manager's capabilities and how to control them.

- **What is the typical volume of data being migrated?** 1 million bytes, 100 million, or several gig of data?
- **How often will data be migrated?** Once every night, or every 15 minutes? A single large load once per month?
- **What time of day or night is optimal for moving the data?** Some sites cannot get access to data until after the nightly production jobs are complete (which may be at 5:30 AM). When is the warehouse required? Are different time zones involved? This complex question impacts not only your performance requirements, but also your scheduling methodology.
- **Is there a key value in the source data that will identify "changed" rows?** ETL Manager can locate changed data using an SQL WHERE criteria, unless an extract file exists with delta changes. Is this file available? Do you know what the WHERE criteria should look like?
- **How many designers will be developing extractions with ETL Manager?** Some sites designate a single ETL Manager administrator who is responsible for all the extract and load routines that are used to develop their data marts. (There are rarely any conflicts in such an environment: one user has complete control over the requests, the source data, and the target locations that will be written.) Other sites have a team of developers interacting to create a collective data warehouse.

Reviewing Your Data Sources

Aspects of your data sources contribute to the performance of ETL Manager. Consider the following issues when reviewing your data sources:

- **Number of columns in an extract.** iWay supports about 400 columns in an SQL command. Previous servers, however, support fewer. If more columns are required, you may need to have multiple requests. Or you can first concatenate columns.
- **Length and actual value of column names.** For information on this topic, see Chapter 10, *Data Management*.
- **Number of tables.** iWay can handle the joining of up to 15 data sources, but the complexity increases as the number of files increases. Thorough knowledge of the data is required to determine if a multi-table join makes sense in any given environment. As a general rule, try to keep the number of tables to a minimum.
- **Location of your tables.** Is a cross-platform (for example, an OS/390 table to a UNIX table) join required? Is all the data in one data source on one platform? Perhaps a cross-DBMS join (for example, C-ISAM to Oracle) is needed? If so, this may impact performance.
- **Data types.** What data is being moved to the warehouse? Are all the data types supported by the current iWay environment? Be sure that the data driver supports the ability to read and extract the data type required by your warehouse. Be sure that the joins you require can be supported by the data types of your columns. Consider the scale (number of decimal points) on floating-point columns, or site-specific columns that are often found in old data sources, such as company-specific date fields that require the use of a home-grown assembler routine.

For more information on data sources see Chapter 10, *Data Management*.

Reviewing Data Targets

The data targets you use for your application affect ETL Manager's performance. Consider the following issues which pertain to the tables or files being written by ETL Manager:

- **Row processing logic.** Is this a total refresh? Will the request be deleting all rows or dropping the table before starting? Is an update or delete necessary? The processing logic is specified in the Workflow tool of the ETL Manager Workbench.
- **Location of the tables.** Are the tables local to the machine where the Server will be running? What connect information will be needed to resolve the names for remote data sources? Can it be handled in the Access File for the iWay Synonym? ETL Manager Workbench allows you to specify these items, but in many cases it is best to allow the iWay Hub Server to manage the location of tables. Review the EDASPROF file for your user ID, as well as the Access File that is used with the Server to establish table names and their storage locations.
- **Data types.** Does iWay support the data types required by your target table? Are there any limitations concerning size or complete write access to certain data values? For more information, see Chapter 10, *Data Management*.

For more information on data targets see Chapter 10, *Data Management*.

Customizing Your Configuration

ETL Manager is an added-value application that uses the technology of the iWay environment to support data migration. The application depends on a properly configured and working iWay configuration. Much of this is verified during the installation and configuration of ETL Manager (and also outlined in your iWay Server documentation). However, there are issues that should be considered.

If only a single ETL Manager administrator will be designing copies, no alterations to the iWay configuration are necessary. However, if the Server will support multiple users, you need to consider the server configuration, the ETL Manager Workbench configuration, and possibly the development methodology (for issues such as naming conventions).

The Server maintains a separate entry in the ETL Manager data store for each user who designs and manages ETL Requests. However, any user can see all ETL Requests on the Server.

Connection, Tools, and Security

You need to consider issues such as your connection, tools, and security options:

- Has the connectivity been checked and verified? Can a Microsoft Windows tool (one is supplied with iWay Connector) make a successful connection to the Server? Are the user IDs available to edit the ETL Manager internal data store? Users designing ETL Requests require SELECT, UPDATE, DELETE, and INSERT authority for the ETL Manager internal tables.
- Are the source definitions correct? Has connectivity to the RDBMS or DBMS sub-system been established? Has access to legacy data files been tested? Can data be retrieved through the Master Files?
- Are the iWay Synonyms correct? Were the Synonyms created through the iWay Catalog Administrator or some other tool? Have they been tested with simple SQL through a generic iWay client tool (such as the one supplied with iWay Connector)?
- Have the security algorithms and connectivity options been verified? Is the inter-machine security trusted or non-trusted? Security is an issue for every iWay environment. Connection options must be correctly specified in the EDASPROF files (see your iWay Server documentation).
- Has the iWay Workspace Web Console been installed, and does the data movement team know how to use it? The iWay Console is a valuable tool for monitoring the run-time processing of an ETL Request.

Naming Target Tables and Target Types

You need to consider the names of target tables and the target types.

- Flat, FOCUS, and Server Transfer files are controlled by ddname, and are independently allocated by the designer. Therefore, they may only create conflict if users select the same file name (operating system dependent) for their output. This can be avoided by establishing a convention for where each user will write their extracted answer sets. For more information, see Chapter 4, *Building an ETL Request*.
- Relational targets with an existing target only require that the correct access rights be given to the approved group of designers. Relational targets with a new target generally default (this depends on the type of data source selected) to assigning an owner ID equal to the logon ID. For more information, see Chapter 10, *Data Management*.

Accessing Target Tables

iWay controls access to local and remote tables using the Synonyms. This description of the data is stored in a directory (a PDS for OS/390) or set of directories that are searched at run time to establish data types in preparation for retrieval.

In a default configuration of the Server, there can only be one “named” iWay Synonym in the general directory for each Server. If only one user is controlling the Server, or a valid naming convention has been implemented, then this will not be an issue. This issue is also avoided if multiple users are using only existing targets, which implies that a single table already exists at the Server. If the potential conflict is unavoidable at your site, there are two alternatives:

- **Separate Synonym directories.** Through user-specific EDASPROF files, you can set up specific Synonym directories for each user. This way there will be no conflict for the creation of new synonyms, and a set of generic or system-wide synonyms can still be shared among the users. Synonyms that are proven can be migrated to the generic directory location. A user level EDASPROF file is a procedure that is executed when a specific user ID connects to the Server. This file is given the name or group (depending on the security package and operating system for your server) for the user ID that is making the connection. In this manner, each developer can connect independently to the Server, share common communication links and file definitions, but still create his or her own unique ETL Manager extracts. This method requires an understanding of the Server, but is ultimately the most flexible. For more details on Synonyms and the EDASPROF file, see the appropriate iWay Server documentation for your platform.
- **Separate Servers for each user.** This solution is easiest to implement, but may be more difficult to administer and manage over a long period of time. A completely separate server is configured for each user, so that each user has his or her own Synonym directory space. The servers can still easily share a common library and adjust Synonym search order in the generic EDASPROF file. However, in this solution, any user can connect to this separate server and be isolated from other developers. No user level profiles are necessary. This is an effective method for deploying multiple Server applications, especially if they have no extracts in common. For example, if data marts are being created for two independent departments in the company, set up each department with its own Server and administrator. This may be most effective for testing and evaluating the behavior of ETL Manager in your environment.

Prototyping

The steps below suggest a path to success with ETL Manager that allows for time to learn its behavior and understand its performance characteristics. These steps assume that ETL Manager has been installed, and that the questions throughout this topic have been given proper consideration. The Server and the Scheduler should be started and running. On some machines, this may involve the starting of several programs. Check the appropriate iWay Server documentation for details.

- Do the tutorial in Chapter 6, *ETL Manager Tutorial*, which provides a complete understanding of the features and capabilities of ETL Manager from the perspective of the data warehouse designer.
- Choose a simple SELECT command that uses a key value to select a small number of rows from just one of your source tables. Select about five columns from this table.
- Use a new target to load data into the local default relational data source. Choose an eight character name for this new table. Turn on the Include Duplicates option in the Target Setup window.
- Monitor the iWay Console.
- Run the ETL Request using Immediate Execution. ETL Request tasks should be visible in the iWay Console.
- When the tasks have completed, go to the ETL Reports folder of the ETL Manager Workbench and check the ETL Server Log. Review the messages that are in the log. Errors will display in this log if something fails during processing.
- Check the clocks for your various computers. The Server Scheduler uses the internal clock at the Server—*not* the internal clock of your PC. Is the server in another time zone? Are the clocks in sync? Such discrepancies often cause confusion.
- Schedule the ETL Request you just tested to run once during the next 15 minutes. Monitor the iWay Console at about the time you expect the request to run; check the ETL Server Log again when it completes. Note the time information that is shown on each row of the ETL Server Log report.

Your path to successful data movement has begun. Alternate target types, new targets, row processing options, and procedures are ready to be used. Check the log after each successful run, and become familiar with the messages that outline an ETL Request's operation.

Improving ETL Manager's Performance

ETL Manager's run-time performance depends on many factors: host platform, network protocol, I/O devices, type of SQL command, and selected DBMS. Depending on your environment, one or more of these factors may be in your control. The following topics outline the options that have the greatest impact on performance.

The issues described here may or may not affect your particular configuration. After careful review and comparison, you may determine that your request will need a lot of internal processing and significant I/O. This may be fine if your nightly incremental updates are low in volume or if you are on a processor where disk access is not a bottleneck. However, if you have only one hour to process your data and the Server is on a small machine with few resources, an alternate approach may be necessary.

In either case, review these performance considerations, and compare them to your environment. For most applications, ETL Manager's default functionality will be sufficient for your data migration needs. If the performance targets cannot be met, these guidelines will have at least provided you with some insight into the location of potential bottlenecks such as network transfer, sort processing, RDBMS load, etc. You may find that combining the Server Transfer File or flat file options of ETL Manager with a bulk load processing may be more appropriate. For details, see Chapter 4, *Building an ETL Request*.

Improving Data Extraction Performance

ETL Manager extracts data from data sources using SQL. The process of applying SQL to heterogeneous data sources is very complex and too broad a subject to be easily outlined here. However, the following are issues which you have control over.

- **Providing qualifiers to the data source.** iWay uses index structures (if available) to retrieve rows in the most efficient manner. Therefore, supply screening conditions in your WHERE criteria for as many columns as possible.

Also, check the *iWay Data Administration Guide* to be sure that critical native DBMS optimization fields (primary keys, secondary indexes, and so on) are correctly defined in the Synonym.
- **Evaluating your joins.** iWay is flexible, enabling similar data source, cross-data source, and even cross-server (sometimes called cross-platform) joins. The tradeoff for this flexibility is the amount of work required of the Server, and the number of intermediate files being created. To minimize the impact of joins, try the following:
 - Join your tables using key columns and indexes.
 - Minimize the number of tables used in a single SELECT command.
 - Specify the *smaller* of the two tables being joined first. See Chapter 4, *Building an ETL Request*, for more information on how to do this.

Using Automatic Passthrough

Automatic Passthrough is the ability of a Server to “pass” or “hand-off” an intact SQL command to a subserver or RDBMS subsystem. In effect, it enables a user to speed up ETL Manager. If a request does not meet the requirements for APT, the Server needs to do additional work. SQL commands must be translated into a format (iWay Data Manipulation Language) that is understood by the underlying (typically non-relational) file system(s), forcing the need for a special work file to be created on the Server platform. ETL Manager waits for this temporary work file to be written and then continues processing by loading the RDBMS table.

Note: When creating a new target, when APT is not used, all columns created in relational tables are created with a length that is a multiple of four.

If a request meets the requirements for APT, ETL Manager uses its “streaming” or “pipeline” methodology. This means no intermediate files are written by ETL Manager, rows are “fetched” directly from the source (effectively “off the wire” for communication scenarios), and are then written to the target. Whether ETL Manager is writing a local flat file or sending dynamic SQL to an RDBMS table, an SQL request that runs in APT mode guarantees the best default performance.

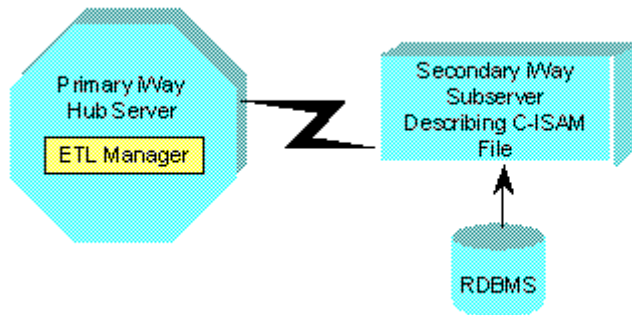
The following environmental and request-based factors most often result in requests failing the requirements of APT:

- Non-relational file structures accessed directly at the Server.
- DEFINE commands in a SUFFIX=EDA Master File or Synonym.
- Cross-platform joins.
- Stop query retrieval selected in immediate execution panel.
- Disable automatic passthrough (APT) is checked in the Output Parameters tab of the Load component in the Workflow tool

For more information on automatic passthrough see the iWay Server documentation for your platform.

Avoiding Non-Automatic Passthrough

If non-APT processing is critical to your extraction process and it is determined to be a bottleneck, it may help to add an additional subserver to your configuration. The critical definition for ETL Manager (and its need to use a HOLD file) is the definition of APT at the server where ETL Manager is installed. For example, if you are reading from a C-ISAM file on UNIX (non-relational), describe the file to a second Server configured expressly to avoid a non-APT condition. Server-to-server shipment of rows may outperform the disk I/O resulting from a HOLD file. This is illustrated in the following figure:



This technique requires that the additional subserver have access to a Master File describing the C-ISAM structure, and that the primary server is configured correctly for communicating with the subserver. An iWay Synonym must be created to point to the correct destination. ETL Manager will use this new synonym for the SQL request instead of the original C-ISAM Master definition (see the Server manual for your platform for more information on configuring iWay Hub and subservers). Once again, the configuration work needed to avoid non-APT may be “over-engineering” in a situation where row volume, disk I/O, or processing power is not an issue.

Another way to avoid non-APT processing is to place your transformations in the Master File of the subserver. This is accomplished by entering a DEFINE in the Master File. This ensures that the calculating and processing work is done directly on the platform where the data resides.

Reference Determining the Use of Automatic Passthrough

You can tell if a request is processed APT or non-APT by looking at the ETL Server Log. A message appears towards the top of the output for every ETL Request, indicating its capacity for passthrough processing.

Reference **Using Real Numbers With Automatic Passthrough**

Real or floating point numbers, also referred to as approximate numeric, are numbers stored in scientific notation.

When iWay retrieves numeric data, it is converted to an intermediate display format. For flat files, FOCUS files, and relational data sources where Automatic Passthrough is not used, the display format is that found in the Synonym. However, when iWay uses Automatic Passthrough, it does *not* use the Master File. In this case, iWay uses a default format of 20.2 (that is, 20 digits with 2 places to the right of the decimal). This format may not be appropriate for data warehousing needs.

You can change the format with an SQL command. You can include this command in the profile on the subserver (where the data source resides), and all real numbers will be rounded to 4 digits.

If different tables in different data sources have different precisions, set the precisions for each ETL Request with a Pre-Extract procedure. If the data source resides at the Server, create a Pre-Extract procedure with this command.

If the data source resides on a subserver, create a Pre-Extract procedure with the following commands

```
SQL EDA SET SERVER subserver
-REMOTE BEGIN
SQL database SET CONVERSION FLOAT PRECISION nn [mm]
-REMOTE END
```

where:

subserver

Identifies the subserver where the data resides.

If one data source has real numbers stored with different scales, address this disparity by either forcing non-Automatic Passthrough or performing separate ETL Requests.

For more information on this command, see the *iWay Data Administration Guide*.

Syntax **How to Change the Format of Real Numbers**

```
SQL database SET CONVERSION FLOAT PRECISION nn [mm]
```

where:

database

Is the type of data source to which you are connected.

nn

Is the precision. This number must be greater than 1 and less than 20.

mm

Is the scale (the number of places to the right of the decimal).

Example **Changing the Format of Real Numbers**

For example, if your data source is DB2, and you want to set the number of digits to the right of the decimal to 4, issue the following command:

```
SQL DB2 SET CONVERSION FLOAT PRECISION 20 4
```

Improving Data Transit Performance

When you perform an ETL Request, you need to move data from your source platform to your target platform. Generally, iWay takes care of this for you. However, if you are moving vast quantities of data, you can improve performance by bypassing iWay and using a dedicated file transport utility, such as FTP.

For more information on configuring ETL Manager to do this, see Chapter 4, *Building an ETL Request*.

Improving Data Loading Performance

You can improve performance when loading data into the data target. Keep the following suggestions in mind. For more information on these options, see Chapter 4, *Building an ETL Request*.

- **Initializing the target table.** Before you load data into an existing target table (in an RDBMS), ETL Manager gives you the option of deleting all rows.
- **Handling duplicates.** ETL Manager offers you several options for handling duplicate entries (reject, update, include, or delete). The fastest option is to include duplicate entries.

If you do need to use one of the other options, try to provide unique keys on the table.

- **Specifying a commit value.** ETL Manager enables you to specify a COMMIT value for loading to an RDBMS. A high commit generally improves performance, but may not be appropriate for your application.
- **Using an alternate loader.** Instead of using ETL Manager to load data into your data target, you can specify an alternate loading mechanism, such as Oracle's SQL*Loader.

Representing a Null Value in a Flat File and With Bulk Load

Previous versions of ETL Manager used a period to represent a null value, which resulted in errors in subsequent processing, including bulk load. To prevent this problem, when a null value is encountered when using a 5.1 Server, a blank is written to a flat file target.

You can change the null value from a blank back to a period or another alternate character with the &HNODATA variable when the target is a flat file.

Procedure How to Set the Null Character to an Alternate Character

Enter the following as a pre-extract procedure:

```
CMOPTION SET, HNODATA, 'char'
```

where:

char

Is the character you wish to represent a null value.

Example Setting the Null Character to a Period

The following sets the null character to a period:

```
CMOPTION SET, HNODATA, '.'
```

Customizing Your Environment

You can control your environment by using the SET command with SET parameters.

The SET command is typically set in a user profile for the Server or in the general EDASPROF file. You can easily test them with your ETL Request by using the CMOPTION procedure as a Pre-Extract procedure.

The following are some of the parameters that affect ETL Manager:

```
MSG={ON|OFF}
```

Turns on iWay messaging for the log.

```
JOINTYPE={SORTMERGE|NESTEDLOOP}
```

Affects SORTMERGE or NESTEDLOOP logic.

`JOINOPT=GMINT`

Affects how SQL is handed to subservers.

`CDN=ON/OFF`

Turns on continental decimal notation for flat file copies.

Syntax **How to Set Parameters in a Profile**

`SET parameter = option`

where:

`parameter`

Is the setting you wish to change.

`option`

Is a valid value for the parameter.

Example **Setting a Parameter in a Profile**

The following sets the CDN SET parameter to ON.

`SET CDN = ON`

Syntax **How to Set Parameters With a Procedure**

`CMOPTION SET, parameter, option`

where:

`parameter`

Is the setting you wish to change.

`option`

Is a valid value for the parameter.

Example **Setting a Parameter With a Procedure**

Including the following as a pre-extract procedure sets the CDN SET parameter to ON.

`CMOPTION SET, CDN, ON`

Creating a DEFINE in the Master File

You can create transforms by creating a DEFINE in the Master File. This creates an extract transform that appears with the list of columns in a source table in the Sources tool.

A DEFINE in the Master File is useful in the following situations:

- **When you want to create static transformations that are constant through an organization.** For example, you might want to create a column of phone numbers with the dashes removed, or a list of Department codes decoded (for example, 1 for Accounting, 2 for Marketing, and so on).
- **When you want all ETL Requests to use a set of common business rules.** For example, you might want to set a definition like $PROFIT = REVENUE - EXPENSES$.
- **When you want to simplify ETL Request maintenance.** If the PROFIT expression noted above ever changes, it need only be edited in one location: at the source Server.

A DEFINE in the Master File can also be used in these situations. However, it is also possible to use an extract transform.

- **When you want to screen based on the results of a transformation.** For example, you might want to screen based on the size of PROFIT.
- **When you want to create transformations based on user routines that are located on the source machine.** Specialized in-house routines, tested over time, may be required for your warehousing application. ETL Manager and your data mart may be located on a UNIX platform—business rules may dictate that an OS/390 COBOL program be executed for each retrieved row. You can exploit this COBOL program using a DEFINE.

It is often useful to prototype the transformations using ETL Manager with a limited number of rows, and then place them in the source definition after analyzing your performance.

Note: If you have an ETL Request that uses the Master File you are creating a DEFINE in, to use the new DEFINE in the Source component, click *Refresh Columns* in the Source window.

Example **Converting Missing Legacy Data Into “Null”**

This example of a transform or DEFINE in the Master File describes how to convert legacy zeros that you would like to represent as null or missing (or be their true values if present). Any “flag” type value might indicate missing data for a particular application.

Suppose the source column in this example is called SOURCE_ZERO. You would use the following syntax:

```
TARGET_ZERO/I4 MISSING ON = IF SOURCE_ZERO EQ 0 THEN MISSING  
                           ELSE SOURCE_ZERO ;
```

Tip: If your source is a Server for OS/390, you can perform this conversion automatically using ALPHACLEAN. For more information, see the appropriate iWay Server for OS/390 manual.

APPENDIX A

ETL Manager Internal Data Store

Topics:

- Log and Statistics Tables
- Maintaining the Server Log and Statistics Table

This topic provides information about the ETL Manager internal data store.

Log and Statistics Tables

ETL Manager stores log information and ETL Request statistics in an internal data source. These tables may be accessed for read-only purposes through the Server. The tables are optimized for use by ETL Manager and are subject to change by Information Builders. This topic briefly describes the format and function for each column of the tables.

The following tables are described by like-named Master Files.

ETLLOG	ETL Request run-time message log.
ETLSTATS	Run-time statistics for each ETL Request.

ETLLOG Table

The ETLLOG table contains the ETL Request run-time message log.

Column	Format	Function
DATE_STAMP	Char 20	Date and time of log entry.
USER_FIELD	Char 8	Request author.
JOBNAME	Char 64	Internal identifier.
REQ_NAME	Char 32	Request name.
COUNTER	Integer	
MESSAGE	Char 80	Message string in the log.

ETLSTATS Table

The ETLSTATS table contains the run-time statistics for each ETL Request.

Column	Format	Function
S_USERID	Char 32	Request author.
S_REQ_NAME	Char 32	Request name.
S_REQ_ID	Char 64	Internal identifier for ETL Request jobs.
STAT_SBM_T	Char 20	Submit date and time for ETL Request job.
STAT_STRT_T	Alpha 20	Start date and time.
STAT_END_T	Alpha 20	End date and time.

Column	Format	Function
STAT_RTRV_T	Alpha 20	Date and time of retrieval from of deferred ETL Request.
STAT_TRANS	Char 15	Rows processed.
STAT_ACCEPTS	Char 15	Rows accepted by ETL Manager.
STAT_NOMATCH	Char 15	New rows rejected by ETL Manager.
STAT_FORMAT	Char 15	Format rejections from ETL Manager.
STAT_INVALID	Char 15	Rows input by ETL Manager.
STAT_DUPLS	Char 15	Number of rejected duplicate records.
STAT_INPUT	Char 15	Number of segments input.
STAT_CHNGD	Char 15	Rows updated.
STAT_DELTD	Char 15	Rows deleted.
REQ_RC	Char 16	ETL Manager internal return code.
ICM_RETURN	Char 12	NLS return code.
TSCOMID	Char 15	TSCOM ID for killing job.
ORIG_REQ	Char 32	Original request for dependency.
ORIG_USER	Char 32	Original user for dependency.
DEP_CNT	Integer	Dependency number.
ETL__ERRTX	Char 80	Request return code for text message.

Maintaining the Server Log and Statistics Table

The Server Log, stored in the ETLLOG table, and the Server Statistics table, stored in ETLSTATS, should be monitored and periodically initialized, depending on the overall usage of ETL Manager. On average, approximately 40 new rows of 120 bytes are added to the log for the successful completion of an ETL Request. However, processing errors can greatly increase this amount. ETLSTATS has a single row per ETL Request, so it is a much smaller table.

The Server ships with two procedures that help you maintain these tables. On a secure Server, these procedures can only be run by the user ID specified as the Administrator when the Server is installed.

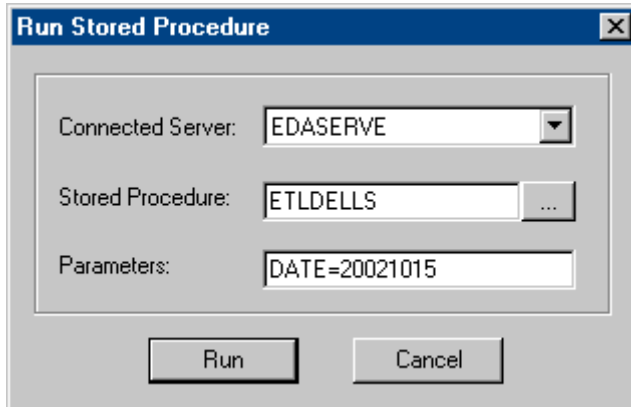
- ETLDELLS deletes rows up to a date that you specify from the log and statistics tables.
- ETLDRPLS drops and recreates the log and statistics tables.

Procedure How to Delete Rows From the Server Log and Statistics Tables Using ETLDELLS

1. In the ETL Manager Workbench, select *Run Stored Procedure* from the Tools menu.
2. Select the Server whose log you want to delete rows from in the Connected Server drop-down list.
3. Enter *ETLDELLS* in the Stored Procedure field.
4. In the Parameters field enter
`DATE=yyyymmdd`
where:
`yyyymmdd`
Is the date up to which you want to delete all rows.
5. Click *Run*.

Example Deleting Rows From the Server Log

If you wanted to delete all rows in the Server Log up to October 15, 2002, you would enter the following information in the Run Stored Procedure dialog box:



Procedure How to Automatically Delete Rows From the Server Log and Statistics Table

It may be useful for a Server to automate deleting rows from the Server log and statistics tables. For example, you can delete all rows more than a week old on a daily basis.

1. Create a stored procedure on the Server with the following lines

```
-SET &RDATE=AYMD (&YYMD, -7, ' I8 ' ) ;
EXEC ETLDELLS .DATE=&RDATE
```
2. Create a new ETL Request. In the Properties section, select *Run Stored Procedures Only (RPCs)*.
3. In the Procedures tool, enter the name of the procedure you created as Pre-Extract procedure #1.

Procedure How to Drop and Recreate the Log and Statistics Tables

1. In the ETL Manager Workbench, select *Run Stored Procedure* from the Tools menu.
2. Select the Server whose log and statistics tables you want to drop and recreate from the Connected Server drop-down list.
3. Enter *ETLCRTLS* in the Stored Procedure field.
4. Click *Run*.

Note: You can also drop and recreate the log and statistics tables in the Web Console. For more details, see Chapter 12, *Using the iWay Web Console*.

APPENDIX B

Problems, Errors, and Troubleshooting

Topics:

- ETL Manager Workbench Status Codes and Server Messages
- Server Run-Time Error Codes
- ETL Management Problem Resolution
- Server Message Codes
- Including Run-Time Messages in the Server Log
- Performing a Trace to Diagnose Server Problems

This topic assists you in resolving problems that may occur during the development of your ETL Manager application.

ETL Manager Workbench Status Codes and Server Messages

There are three types of errors or messages that can occur in the ETL Manager Workbench:

- **iWay Connector communication errors.** These error conditions usually indicate a problem communicating with the server or network.
- **Internal ETL Manager Workbench errors.** If these errors are self-explanatory, you are given the option of choosing OK or Cancel. If not, the error may indicate problems in memory or with the environment in which ETL Manager Workbench is operating.
- **Messages from the Server.** The most common messages are those that are sent by the Server. These will display in the message window of the ETL Manager Workbench. Along with descriptive text, there will be a message origin and a message code. The following is the beginning of the message

(XXXnnnn)

where:

XXX

Is the message origin.

nnnn

Is the message code.

Messages of this type may come directly from the iWay subserver during the retrieval of catalog information or rows from a source table, or from the Server as administrative messages.

Common codes of the above three types are described in the following topics. Also see the appropriate iWay Server documentation for your platform, and the Server codes at the end of this topic. For information on message filtering, see Chapter 4, *Building an ETL Request*.

Server Messages

Some common server messages are listed below:

EDA054. EDA Security error

There is security on the Master File, and the password has not been set correctly. This should be done in the EDASPROF or the user profile for the server.

EDA14010 -column-name- IS NOT A VALID COLUMN NAME

ETL Manager Workbench issues a PREPARE for the SQL command, and the column name does not match the Master File. If you have edited the Master File at the source, you may need to create the Synonym with Automatic Passthru turned off.

EDA1006 THE DDNAME REQUESTED IS NOT ALLOCATED OR DLBL'ED : table
EDA226 FILEDEF OR ALLOC MISSING FOR EXTERNAL FILE: table
EDA1400 SQLCODE IS : 1006/000003EE
EDA1405 SQL PREPARE ERROR

These messages, or some combination of them, can display when an Synonym for a source file exists but the file it describes is not allocated on the server. These errors can occur upon exiting from the Query Builder or upon an attempt to run a Test Query. They can also occur when using Metadata Manager or some other tool to create a synonym.

FOC1538 4GL procedure doesn't exist: proc

One of the procedures required by the Server could not be found. This may indicate that a new version of the server is needed for full compatibility with the client version being used.

FOC35804 BAD STATUS CODE (-9) RECEIVED FROM SERVER subserve

The Server was unable to connect to a subserver listed in the server's configuration file (ODIN.CFG).

For more information on error messages, see the ETL Manager online help.

iWay Connector Communications Errors

If there is a problem connecting to a Server, a message is displayed in the message window. Following are some of the more common errors:

Connect Failed. Return code = -17 Failure to connect to the server
because of an invalid user name or password

The user ID and/or password that you entered was invalid for the selected server.

Connect Failed. Return code = -9 (Unable to connect to server)

The selected server was unavailable; it may not be running. The iWay Connector configuration (ODIN.CFG) may be incorrect.

Server Run-Time Error Codes

The errors listed here will usually display in the Server Log unless otherwise noted. Several of these are formal Server error codes. You can see a complete list in *Server Message Codes* on page B-6. Other errors are related to the iWay environment, operating system, or specific data source system being accessed. Additional references from these resources may be necessary.

ICMnnnn

These are administrative messages received during processing. They illustrate the progress of the ETL Request as it proceeds through its various steps.

CMYnnnn

This is generally the last, or nearly the last, summary message in the log.

FOC282 -or EDA282 RESULT OF EXPRESSION IS NOT COMPATIBLE WITH..

This error generally occurs on a transformation string or on a DEFINE in a source Master File. It means that the answer for a calculated field cannot be placed into the column name on the left-hand side of the equal sign. Typically, this is caused by an attempt to move an alpha field directly into a numeric field (or vice versa), or a character string into a column whose format is less than the length of the desired string. For details on this and other errors in this category, see the appropriate FOCUS documentation for your server platform.

FOC14010 -column-name- IS NOT A VALID COLUMN NAME

ETL Manager issued a prepare for the SQL command, and the column name does not match the Master File. The Master File may have changed since the ETL Request was designed. For Server Transfer Files, another possibility is that the Synonym at the Server does not match the Master File at the source. See Chapter 4, *Building an ETL Request*, for more information. If you have edited the Master File at the source, you may need to create the Synonym with Automatic Passthru turned off.

FOC35804 BAD STATUS CODE (-12) RECEIVED FROM SERVER server

FOC35862 EDA STATUS (-12): ERROR RECEIVING DATA FROM SERVER

These errors indicate that something has gone wrong at the server identified in the message.

One common reason for this error is that the server is running out of disk space. Check the location of the NT or UNIX temporary directory, CMS temporary minidisk, or OS/390 temporary data set allocation.

On OS/390, another possibility is that the data sets allocated to EDASYNA and EDASYNM are full. If so, delete unnecessary Master and Access numbers, and compress the data sets.

ETL Management Problem Resolution

Situational errors sometimes occur in environments with complex iWay configurations or multiple users. The list below is meant to provide you with error messages similar to what you may encounter, and offer you resolutions such as you may find in ETL Manager. Each message is characterized by a single symptom, followed by an explanation or references to other materials.

Unique Constraint Error... Duplicate keys not permitted... or similar error for any of the relational database targets. ETL Manager has duplicate values in its extracted answer set, and there is a unique key on the table. For new targets, a unique key value may need to be specified in the Target Setup window. For existing targets, another column may need to be added to a concatenated key.

This error may also be caused by the retrieval mechanism: the use of the DISTINCT operator in the SQL command may really be the desired result (more rows than expected are being retrieved), or the source data may be a complex hierarchy or join that is creating duplicate rows. See the usage notes for your specific data target.

Format errors caused by “dirty” or invalid data. Legacy data is not always consistent in format and type. Older systems may have stored standard internal packed numbers for revenue figures but handled special logic conditions by inserting characters or low-values as needed by the application program. These issues often require handling with DEFINES in the Master File at the source. There is a complete set of subroutines to handle various bit and byte oddities in your host information. For more information see Chapter 4, *Building an ETL Request*, and the appropriate FOCUS documentation for your platform. Problems of this type usually make themselves known through data exceptions and other error conditions (these would display in the Server log). Depending on the data type combinations, they may also result in rejected records or a misinterpretation of missing data.

Confusing answer set; too many rows as the result of a Cartesian product. The rules of SQL control the join process for the source data to ETL Manager. Many-to-many joins result in a Cartesian product. This may be what the warehouse requires, but very often it is confusing because it results in additional rows being generated. The Cartesian product may also result from the extraction of columns that cross two paths of a multi-path legacy hierarchy. This, in fact, becomes a join between two relational tables and can result in a confusing answer set. If this occurs, reduce the number of joins, and/or access only one path of a complex hierarchy. For details on the Cartesian product, see the appropriate iWay Server documentation for your platform.

Server Message Codes

These are message codes from the Server that display in the Server Log, ETL Workbench, or iWay Web Console. These messages all have a message origin of ICM.

Code	Definition
18054	Issuing DROP TABLE for %1. For FOCUS and FUSION targets a DROP TABLE command is issued.
18056	Issuing CREATE TABLE for %1. A CREATE TABLE command for the target table is issued to the RDBMS.
18076	ETL request: %1 - finished processing. The ETL request has finished processing.
18122	ETL request - %1 (Owner: %2)submitted. The ETL request has been submitted for the owner.
18162	Error saving request %1 for User %2. ETL Manager was unable to save the request. The disk may be full, or the user may not have permission to write to the disk.
18164	Request %1 created/updated successfully for User %2. ETL Manager saved the request for the specified user.
18171	Complete deleting: %1 for User: %2. The ETL request has been deleted.
18193	Error deleting request %1. ETL Manager was unable to delete the specified request. It may have been deleted by another process, or the user may not have permission to write to the directory where the request is stored.
18237	Ending bulk load procedure. The bulk load procedure has completed.
18292	Starting %1 procedure. The transport (FTP) procedure is starting.
18303	Create synonym executed for: %1. ETL Manager is creating a synonym for the specified data target.
18354	Migration procedure completed successfully. The migration of ETL Requests from a 4.3 or 5.1 server has completed successfully.
18372	-- stats for %1 ----- . The following lines contain statistics for the specified target table.
18429	PREPARE executed. This determines if the SQL SELECT statement is valid, and whether Automatic Pass Through can be used.

Code	Definition
18440	Request will process data via NON-Pass Through (NON-APT). Automatic Pass Through could not be used.
18451	HOLD file executed. Output file named: %1. A HOLD file was created with the specified name.
18473	Attempting to run Stored Procedure: %1. ETL Manager is attempting to run the procedure specified in the request.
18514	%1 : %2 accepted by target table. The specified number of rows or segments were accepted by the target table.
18515	%1 : %2 inserted into target table. ETL Manger attempted to insert the specified number of rows or segments into the target table.
18516	%1 : %2 processed by ETL job. ETL Manager processed the specified number of rows or segments.
18517	%1 : %2 updated in target table. The specified number of rows or segments were updated in the target table.
18518	%1 : %2 deleted from target table. The specified number of rows or segments were deleted from the target table.
18519	%1 : %2 rejected due to format errors. The specified number of rows or segments were rejected.
18520	%1 : %2 rejected due to validation errors. The specified number of rows or segments were rejected.
18521	%1 : %2 rejected due to no match logic. The specified number of rows or segments were rejected.
18522	%1 : %2 rejected because duplicate %3 exists. The specified number of rows or segments were rejected.
18528	ETL Request name (REQ_NAME) Cannot be Blank. When initiating a request using CMRUN or CMASAP, the request name must be specified.
18529	Stop! Request not found for %1 : %2. The specified ETL Request was not found on the Server
18561	ETL Request %1/%2 Not submitted. The ETL Request was not submitted.
18688	ETL Manager scheduler stopped. The ETL Manager Scheduler has stopped.

Code	Definition
18701	Bulk Insert active. INSERTSIZE = %1. A relational databases bulk insert API is used with the insertsize as specified.
18741	%1 type %2 New Table. The ETL Request creates a new data target.
18742	%1 type %2 Existing Table. The ETL request writes to an existing data target.
18743	Starting Load. The process of loading records into the data target has started.
18744	Ending Load. The process of loading records into the data target has ended.
18760	Skipping %1 row(s) before starting load. The ETL Request was initiated with Start at row specified.
18761	Stopping load after %1 row(s). The request was initiated with Stop at row specified.
18762	ETL Job ID: %1. The ETL Manager generated job ID follows.
18763	ETL Request %1 complete. The ETL Request completed successfully.
18764	ETL Request %1 failed; RC = %2. The ETL Request failed with the return code indicated.
18766	ETL Request does not exist. Please refresh the list of requests. The selected request no longer exists. Another user may have deleted it.
18767	Failed ETL Request %1. Skipping dependencies. The ETL request failed, and dependent requests are not run.
18768	Migrate procedure failed. The migration procedure failed.
18769	No path for CMSCHED DB to migrate. No path was specified for a prior version ETL requests. Enter the directory that contains the Master File Description or Synonym CMSCHED.MAS.
18770	Error to send email: RC = 1%. An error occurred when sending e-mail.
18771	Email sent to %1. The e-mail was sent.
18772	Error to send email: connection failed: RC = %1. A connection failure occurred while sending e-mail.
18773	Error run deferred request %1. An error occurred while running the deferred request.

Code	Definition
18774	Deferred request %1 created. A deferred request is created.

Including Run-Time Messages in the Server Log

The Dialogue Manager command `-TYPE` can be used to display messages. This command can be included in a procedure. If this command is used in your procedure, the text will display in the Server Log File along with all other status and administrative messages for an ETL Request. The text will receive a time stamp as recorded by the previous formal administrative message that displayed in the log.

Syntax How to Generate a Message Including a Time Stamp

Use the following syntax inside your procedure every time you want a message with a unique time ID:

```
-TYPE Messages from procedure
```

Performing a Trace to Diagnose Server Problems

When you receive a server error message, you can perform a trace to find the source of the problem.

Procedure How to Perform a Trace

1. In the iWay 52 ETL Manager Server Start menu program folder, select *Start With Traces* from the Alternate Server Startups menu to start the server with traces.
2. Reproduce the problem you encountered.
3. In the iWay 52 ETL Manager Server Start menu program folder, select *Command Window for Manual Operations* from the Diagnostic Functions menu to open the command window.

4. Issue the following commands:

```
edastart -stop
edastart -savediag
```

A sub-directory is created under your configuration directory, and a message displays the location of the sub-directory. For example:

```
savendir = D:\ibi\srv52\etl\sd123456
```

5. Create a .zip or .tar file that contains the contents of this directory, and upload it to Information Builders.

Index

A

- Access Files
 - migrating 12-15
 - viewing 8-7
- aggregating columns 4-34
- alphanumeric fields 4-73
- alternate files 4-92
 - bulk loading 4-92
- Alternate Files tab 4-92
- alternate load options 4-90
- amper variables
 - CMOPTION 8-8
 - CMOPTION procedure 8-7
- API programs 5-15
 - executing ETL Requests 5-15
- APP PATH 12-7
 - configuring 12-7
- application directories
 - creating 12-8 to 12-9
 - name spaces 10-2
- application paths
 - APP PATH 12-7
 - restricting access 11-2 to 11-3
- applications 13-2
 - designing 13-2
- automapping 4-56
- automatic passthrough 13-9 to 13-10
 - avoiding 13-10
 - real numbers and 13-11

B

- building ETL Requests 4-5
- Bulk Load tab 4-89 to 4-92
 - Alternate Files tab 4-92
 - DB2 Options tab 4-95, 4-97
 - Nucleus Options tab 4-95, 4-99
 - Oracle Options tab 4-95 to 4-96
 - Record Logging tab 4-102 to 4-103
 - Run Options tab 4-93 to 4-94
 - Target Database tab 4-90 to 4-91
 - Teradata Options tab 4-95, 4-100
- bulk loading 4-89 to 4-90, 4-102
 - additional options 4-95 to 4-97, 4-99 to 4-100
 - alternate files 4-92
 - command files 4-101
 - control files 4-101
 - data target 4-90 to 4-91
 - record logging 4-102 to 4-103
 - requirements 4-89 to 4-90
 - run options 4-93 to 4-94

C

- CMLOG2 table A-2
 - Server Log A-4 to A-5
- CMOPTION procedure 8-7 to 8-8
 - amper variables 8-7 to 8-8
 - SET parameters 8-7 to 8-8
- CMRUN procedure 5-8
 - executing ETL Requests 5-8 to 5-12
 - return codes 5-8, 5-12
- CMSTATS table A-2
 - Server Statistics table A-4 to A-5

- columns 4-30, 6-12, 6-15
 - aggregating 4-34
 - Columns component 4-30, 6-12, 6-15
 - extract transforms 4-34
 - naming 10-7 to 10-8
 - selecting 4-30, 6-12, 6-15
- Columns component 4-30, 6-12, 6-15
 - Columns window 4-30 to 4-31, 6-12, 6-15
- Columns window 4-30 to 4-31, 6-12, 6-15
- command files 4-101
 - bulk loading 4-101
- commands
 - DEFINE 13-15
 - SQL 4-42
 - TYPE B-9
- complex mapping 4-56 to 4-57
- components 4-7
 - adding 4-7
 - closing 4-8
 - connecting 4-7
 - deleting 4-7
 - disconnecting 4-7
 - maximizing 4-8
 - opening 4-8
 - selecting fields 4-8
- Components box 4-6
- Configure Application Path window 12-9
- configuring ETL Manager 2-1 to 2-2
- configuring the Server 2-2
- connection to Server 13-5
- control files 4-101
 - bulk loading 4-101
- creating joins
 - Join calculator 4-21
- CSMASAP procedure 5-13
 - executing ETL Requests 5-13 to 5-14

- Customer Support Service 1-v
 - How to contact 1-v
 - Information required 1-v

D

- data 10-1
 - data sources 10-2
 - data targets 10-4
 - Master Files 10-8
 - Synonyms 10-8
- data loading 13-12
- data migration 1-2
- data source columns 4-13
 - updating 4-13
- data sources 10-2
 - column names 10-4
 - columns 4-30, 6-12, 6-15
 - joining 4-17, 6-8
 - naming 10-4
 - performance 13-3
 - reviewing 13-3
 - selecting 4-9, 6-2
 - updating 4-13
- data store A-1
 - tables A-2
- data store tables A-2
 - CMLOG2 A-2
 - CMSTATS A-2
- data target columns 4-53
 - updating 4-53
- data target options
 - bulk load 4-89
 - Bulk Load tab 4-89
 - output parameters 4-83, 4-85 to 4-86, 4-88
 - Output Parametes tab 4-83, 4-85 to 4-86, 4-88
 - target setup options 4-77 to 4-78
 - Target Setup tab 4-77 to 4-78

- data target options (*continued*)
 - transport options 4-44
- data targets 10-4
 - accessing 11-2, 13-6
 - bulk loading 4-90 to 4-91
 - column names 10-7 to 10-8
 - existing 4-46, 4-49, 4-51 to 4-52
 - flat files 4-46 to 4-47
 - local non-relational 10-6
 - local relational 10-5
 - naming 10-7, 13-5
 - new 4-46, 4-48
 - non-relational 4-46 to 4-47
 - options 4-77, 6-24 to 6-25
 - performance 13-4
 - relational 4-46 to 4-47
 - remote destinations 10-7
 - reviewing 13-4
 - security 11-2
 - selecting 4-46, 4-48 to 4-49, 4-51 to 4-52, 6-19
 - Server Transfer Files 4-46 to 4-47, 10-6
 - updating 4-53
 - validating records 6-19
- data transit 13-12
- data types 13-5
- data warehousing 1-2
 - designing applications 13-2
 - performance 13-1
 - prototyping 13-7
- date fields 4-73 to 4-74
 - legacy character fields 4-76
- DB2 Options tab 4-95, 4-97
- DEFINE command 13-15 to 13-16
- dependencies 5-5, 5-7
 - guidelines 5-5
 - scheduling 5-5, 5-7

- Dependencies tool 5-5, 5-7
 - Dependencies window 5-5 to 5-6

- Dependencies window 5-5 to 5-6

- Dialogue Manager commands
 - TYPE B-9

E

- EDAGET procedure 8-7

- Edit Condition calculator 4-22, 4-25

- editing SQL commands 4-42

- error messages B-3
 - run-time B-4
 - server B-3

- errors B-1
 - resolutions B-5
 - run-time B-3
 - Server B-3, B-6

- ETL Configuration window 12-5

- ETL management 1-2

- ETL Manager 1-1 to 1-3
 - components 1-3
 - configuration 13-4
 - configuring 2-1 to 2-2, 12-3 to 12-5
 - configuring behavior 12-1 to 12-2
 - help 3-8
 - installing 2-1 to 2-2
 - iWay Web Console 12-1 to 12-2
 - messages 3-6
 - processing 1-3
 - Server 1-5
 - upgrading 2-11

- ETL Manager components 1-3
 - ETL Workbench 1-3
 - iWay Hub Server 1-3

- ETL Manager DB2 database configuration report 9-2

- ETL Manager processing 1-3
- ETL Manager Workbench
 - installing 2-3 to 2-4
 - removing 2-3
- ETL Request log 6-28
 - viewing 6-28
- ETL Request properties 4-2
 - Properties window 4-4
 - setting 4-2
- ETL Requests 1-3, 7-1
 - agent information 12-12 to 12-13
 - building 4-1 to 4-2, 4-5, 6-1 to 6-2
 - cancelling 12-10
 - controlling 8-2, 8-4 to 8-5
 - copying 7-4 to 7-5
 - creating 4-1 to 4-2, 6-1 to 6-2
 - data sources 4-9, 6-2
 - data targets 4-46, 4-48 to 4-49, 4-51 to 4-52, 4-77, 6-19
 - deferred request information 12-12
 - deleting 7-5 to 7-6, 12-12
 - dependencies 5-5, 5-7
 - editing 7-2
 - executing 6-27 to 6-28, 7-7
 - exporting 7-6
 - failed 5-5
 - field formats 4-72 to 4-73
 - filtering 4-22, 6-10
 - filtering display of 7-9
 - mappings 4-56, 6-19
 - migrating 7-4 to 7-5, 12-14, 12-16 to 12-17
 - printing 7-4
 - procedures 8-2, 8-4 to 8-5
 - properties 4-2, 7-8 to 7-9
 - recovering 5-5
 - reports 9-2
 - restarting 5-5
 - running 12-11
 - running deferred 12-11 to 12-12
 - running remotely 5-7
- ETL Requests (*continued*)
 - saving 7-3
 - scheduling 5-1 to 5-5, 5-7, 6-27
 - security 11-2
 - sorting 4-39 to 4-40
 - starting 4-2
 - status 5-18
 - storing 10-2
 - testing SQL 4-42, 6-17
 - viewing 12-11
 - viewing log 12-13
 - viewing message logs 12-14
 - viewing run-time information 12-12
 - viewing statistics 12-14
 - Workflow tool 4-6
 - working with 12-10
- ETL Server Log (details) report 9-2
- ETL Server. See Server
- ETL Workbench 1-3 to 1-4, 3-1
 - components 1-4
 - interface 3-2
 - main toolbar 3-3
 - main window 3-2
 - Server 3-4
 - sessions 3-8
 - Workflow tool 4-6
- ETL Workbench components 1-4
 - Dependencies tool 1-4
 - Procedures tool 1-4
 - Scheduler tool 1-4
 - Workflow tool 1-4
- ETL Workbench session 3-8
- event-based scheduling 5-7
 - API programs and 5-15
 - CMASAP procedure 5-13 to 5-14
 - CMRUN procedure 5-8 to 5-12
 - FOCUS and 5-17
 - status 5-18
 - WebFOCUS and 5-17

- executing ETL Requests 6-27 to 6-28, 7-7
 - Immediate Execution dialog box 7-8
 - immediately 7-7
 - procedures 8-2
 - scheduling 5-2

- expressions
 - functions and 4-65
 - order of evaluation 4-64

- extract transforms 4-13 to 4-14
 - calculations 4-34
 - constant values 4-34 to 4-35
 - creating 4-13 to 4-14, 4-34 to 4-35
 - date values 4-34 to 4-35
 - Extract Transforms calculator 4-15
 - SQL calculator 4-34, 4-36
 - SQL functions 4-37 to 4-38
 - testing 4-15
 - time values 4-34 to 4-35

- Extract Transforms calculator 4-15

- extracting data 13-8
 - performance 13-8

F

- field formats 4-72 to 4-73
 - alphanumeric fields 4-73
 - date fields 4-73 to 4-74
 - formatting new fields 4-73 to 4-75
 - numeric fields 4-73, 4-75

- Filter component 4-22, 6-10
 - Edit Condition calculator 4-22, 4-25
 - Filter window 4-22, 4-24, 6-10

- Filter window 4-22, 4-24, 6-10

- filtering 4-22, 6-10
 - Edit Condition calculator 4-22, 4-25
 - ETL Request display 7-9
 - sub-queries 4-28
 - variables and 4-28 to 4-30

- flat files 4-46 to 4-47

- FOCUS 5-17
 - executing ETL Requests 5-17

- FTP 4-44

- functions 4-65

I

- Immediate Execution dialog box 7-7 to 7-8

- installing ETL Manager 2-1 to 2-2

- installing ETL Manager Workbench 2-4
 - removing previous version 2-3
 - requirements 2-3

- iWay Connector
 - installing 2-2

- iWay Connector messages B-3

- iWay Hub Server 1-3

- iWay Web Console 12-1
 - agent information 12-12 to 12-13
 - application path 12-7
 - cancelling ETL Requests 12-10
 - Configure Application Path window 12-9
 - configuring ETL Manager 12-3 to 12-4
 - controlling Scheduler 12-10
 - deferred request information 12-12
 - deleting ETL Requests 12-12
 - ETL Configuration window 12-5
 - ETL Request log 12-13
 - ETL Requests 12-10
 - message logs 12-14
 - migrating ETL Requests 12-14, 12-16 to 12-17
 - opening 12-2
 - refreshing 12-10
 - running deferred ETL Requests 12-11 to 12-12
 - running ETL Requests 12-11
 - starting 12-2
 - statistics 12-14
 - viewing ETL Requests 12-11

iWay Web Console (*continued*)

viewing run-time information 12-12

J

Join calculator 4-21

Join component 4-17 to 4-18, 6-8

Join window 4-17 to 4-18, 4-20, 6-8

Join window 4-17 to 4-18, 4-20, 6-8

joining data sources 4-17, 6-8

Join component 4-17 to 4-18, 6-8

performance considerations 4-17 to 4-18, 6-8

L

legacy character fields 4-76

converting 4-76

Load component 4-77, 6-24 to 6-25

Load window 4-77

load options 4-90

Load window 4-77, 6-24 to 6-25

Bulk Load tab 4-89

Output Parameters tab 4-83

Record Logging tab 4-102 to 4-103

Target Setup tab 4-77, 4-80

local non-relational data targets 10-6

local relational data targets 10-5

log file 9-1, 9-8

viewing 6-28, 9-4, 9-7 to 9-8, 12-13

M

main toolbar 3-3

main window 3-2

mappings 4-56 to 4-57, 6-19

automapping 4-56, 6-19

complex mapping 4-56 to 4-57

DEFINE in Master File 13-15

field formats 4-72 to 4-73

simple mapping 4-56 to 4-57

temporary columns 4-60

testing 4-57

Transform calculator 4-60

Master Files 10-8

DEFINE command 13-15 to 13-16

mappings 13-15 to 13-16

migrating 12-15

viewing 8-7

Message Log window 3-6

message logs

viewing 12-14

messages 3-6

errors B-6

iWay Connector communications B-3

Message Log window 3-6

Oracle 10-4

run-time B-4, B-9

Server B-2

server B-3

N

name spaces 10-2

non-relational data targets 4-46 to 4-47

Nucleus Options tab 4-95, 4-99

numeric fields 4-73, 4-75

O

Oracle 10-4
 messages 10-4
 Oracle Options tab 4-95 to 4-96
 order of evaluation 4-64
 Output Parameters tab 4-83, 4-85 to 4-86, 4-88

P

passwords 5-13, 11-2
 supplying with procedures 5-13
 performance 13-1
 automatic passthrough 13-9
 data loading 13-12
 data transit 13-12
 extracting data 13-8
 improving 13-8
 SET command 13-13
 procedures 8-1
 CMOPTION procedure 8-7 to 8-8
 EDAGET 8-7
 ETL Requests 8-2, 8-4 to 8-5
 executing 8-2, 8-4 to 8-6
 Procedures tool 8-2, 8-4
 running remotely 8-5 to 8-6
 supplying passwords 5-13
 viewing 8-7
 viewing Access Files 8-7
 viewing Master Files 8-7
 Procedures tool 6-2, 8-2, 8-4
 Procedures window 8-2, 8-5
 Procedures window 8-2, 8-5
 Properties window 4-4

R

real numbers 13-11 to 13-12
 automatic passthrough and 13-11 to 13-12
 record logging 4-102 to 4-103

Record Logging tab 4-102 to 4-103
 relational data targets 4-46 to 4-47
 remote destination data targets 10-7
 Report Browser toolbar 9-5
 reports 9-1 to 9-2, 9-4
 copying 9-6
 ETL Manager DB2 database configuration report 9-2
 ETL Server Log (details) report 9-2
 fonts 9-6
 printing 9-6
 Report Browser toolbar 9-5
 running 9-2 to 9-3
 saving 9-5
 Scheduler Events report 9-2
 Scheduler Log File report 9-2
 Statistics report 9-2

return codes 5-8
 CMRUN procedure 5-8, 5-12
 run options 4-93
 bulk loading 4-93 to 4-94

Run Options tab 4-93 to 4-94

S

Scheduler
 starting 12-10
 stopping 12-10
 Scheduler Events report 9-2
 Scheduler Log File report 9-2
 Scheduler tool 5-2, 6-2, 6-27
 Scheduler window 5-2 to 5-3, 6-27
 Scheduler window 5-2 to 5-3, 6-27
 scheduling ETL Requests 5-1
 Dependencies tool 5-5, 5-7
 event-based scheduling 5-7 to 5-13, 5-15, 5-17 to 5-18

Index

- scheduling ETL Requests (*continued*)
 - failed 5-5
 - Scheduler tool 5-2 to 5-4, 6-27
 - security 11-1
 - customizing 13-5
 - data targets 11-2
 - ETL Request execution 11-2
 - user IDs 11-2
 - Server 1-5
 - configuring 2-2
 - connecting 3-4
 - error codes B-6
 - messages B-2
 - properties 3-5
 - run-time errors B-4
 - Server Properties dialog box 3-5
 - statistics table A-4
 - Server Log A-4, B-9
 - CMLOG2 table A-4 to A-5
 - server messages B-2 to B-3
 - Server Properties dialog box 3-5
 - Server statistics table A-4 to A-5
 - CMSTATS table A-4 to A-5
 - Server Transfer Files 4-46 to 4-47, 10-6
 - SET command 13-13 to 13-14
 - SET parameters 13-13
 - SET parameters 13-13 to 13-14
 - setting with CMOPTION 8-7 to 8-8
 - setting parameters 13-13 to 13-14
 - simple mapping 4-56 to 4-57
 - Sort component 4-39
 - Sort window 4-39, 4-41
 - Sort window 4-39, 4-41
 - sorting 4-39 to 4-40
 - Sort component 4-39
 - Source component 4-9, 6-2
 - Source window 4-9, 6-2
 - Source window 4-9 to 4-10, 6-2
 - SQL calculator 4-34, 4-36
 - SQL functions 4-37
 - SQL commands 4-42
 - editing 4-42
 - testing 4-42, 6-17
 - SQL functions 4-37 to 4-38
 - SQLAnswerSet component 4-42, 6-17
 - SQLAnswerSet window 4-42 to 4-43, 6-17
 - SQLAnswerSet window 4-42 to 4-43, 6-17
 - statistics
 - viewing 12-14
 - Statistics report 9-2
 - status codes B-2
 - storing ETL Requests 10-2
 - sub-queries 4-28
 - Synonyms 10-8
 - guidelines 10-8 to 10-9
- ## T
-
- Target component 4-46, 4-48 to 4-49, 4-51 to 4-52, 4-56, 6-19
 - Target window 4-46, 4-53, 6-19
 - Target window (mappings) 4-56, 4-58, 6-19
 - Target Database tab 4-90
 - Target Setup tab 4-77 to 4-78, 4-80
 - Target window 4-46, 4-53, 6-19
 - Target window (mappings) 4-56, 4-58, 6-19
 - temporary columns 4-60
 - creating 4-60
 - Transform calculator 4-60

Teradata Options tab 4-95, 4-100

testing mappings 4-57

testing SQL 4-42, 6-17

Transform calculator 4-60 to 4-61

functions 4-65

order of evaluation 4-64

transport options

FTP 4-44

Transport window 4-44

Transport window 4-44 to 4-45

troubleshooting B-1

server messages B-3

-TYPE command B-9

U

upgrading ETL Manager 2-11

user IDs 11-2

security 11-2

V

variables 8-7

filtering and 4-28 to 4-30

W

WebFOCUS 5-17

executing ETL Requests 5-17

Workbench. See ETL Manager Workbench

Workflow tool 4-6, 6-2

Components box 4-6

Filter component 4-22, 6-10

Load component 4-77

Sort component 4-39

Source component 4-9, 6-2

SQLAnswerSet component 4-42, 6-17

Target component 4-46, 4-48 to 4-49, 4-51
to 4-52, 4-56 to 4-57, 6-19

workflow tool

Transport component 4-44

Reader Comments

In an ongoing effort to produce effective documentation, the Documentation Services staff at Information Builders welcomes any opinion you can offer regarding this manual.

Please use this form to relay suggestions for improving this publication or to alert us to corrections. Identify specific pages where applicable. You can contact us through the following methods:

Mail: Documentation Services - Customer Support
Information Builders, Inc.
Two Penn Plaza
New York, NY 10121-2898

Fax: (212) 967-0460

E-mail: books_info@ibi.com

Web form: <http://www.informationbuilders.com/bookstore/derf.html>

Name: _____

Company: _____

Address: _____

Telephone: _____ Date: _____

E-mail: _____

Comments:

Reader Comments